A SIMULATION OF HELICOPTER AIRCRAFT IN AN ARMED RECONNAISSANCE MODE, FOR THE CDC 1604 DIGITAL COMPUTER

RICHARD RALPH JOHNSON
AND
NEIL RAY VAN LEEUWEN

[redacted]

[redacted] [redacted]

A SIMULATION OF HELICOPTER AIRCRAFT

IN AN ARMED RECONNAISSANCE MODE, FOR

THE CDC 1604 DIGITAL COMPUTER

by

Richard Ralph Johnson

Captain, United States Marine Corps

B.S., Illinois Institute of Technology, 1957

and

Neil Ray Van Leeuwen

Captain, United States Marine Corps

B.S., University of Utah, 1956

Submitted in partial fulfillment

for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

UNITED STATES NAVAL POSTGRADUATE SCHOOL

May 1966

ABSTRACT

A model is presented which is a computer simulation of a duel involving two helicopter sections, a scout and an attack section, and an armored mobile land target. Terrain features are considered in the model by using a "least square" polynomial to represent the terrain environment. The model was constructed in an attempt to include the possible effects of terrain on tactics used by the combatants in the duel. The computer program, logic and model results are included.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

1. Introduction

The model presented in this thesis, ARMREC, is a time step computer simulation of an armed reconnaissance flight of two helicopter sections in search of a small stationary or moving armored land target. Simultaneous and independent movement of the participants is provided for in the model and all movement takes place over terrain simulated by a "least square" polynomial. The necessary data for the terrain simulation in ARMREC is supplied by a separate program, TERRAIN. The program, TERRAIN, is described in APPENDIX IV. The flight paths of the helicopter sections in the model are either preplanned, and as such are included with the necessary input data for the model, or they are generated as nap of the earth, NOE, flight paths by the ARMREC program.

The interactions included in the ARMREC program, i.e., the search and destroy mission of the helicopters and the ensuing duel between the helicopters and the target, are designed to test the efficiency of a helicopter anti-armored weapons system. The important features of the model are its ability to simulate terrain such that terrain effects on tactical changes by the participants are realistic, and its ability to move any number of participants as independent units over the terrain.

The authors' interest in the possible use of helicopters in an anti-armored role was developed during their 1965 summer operational tour with the Stanford Research Institute research group which supports the Army Combat Developments Command Experimentation Command, CDCEC, at Fort Ord, California.

The authors wish to thank the staff of the Stanford Research

Institute group at Fort Ord, and specifically Mr. Richard Ferris, the Project Scientist, and the members of Team IV on the CDCEC staff for giving them the opportunity to assist in formulating proposed tactics for helicopters in an anti-armored environment. The authors also express their appreciation to Professor Alvin F. Andrus who provided the encouragement and help necessary in completing this thesis.

2. Design of the Model

ARMREC was programmed in FORTRAN- 63 for the CDC-1604 computer and consists of a time step movement of two helicopter sections involved in a search and destroy mission against an armed mobile unit. The model is designed so that Monte Carlo techniques can be used to generate expected outcomes of various tactics used in anti-armored reconnaissance missions which may be assigned to helicopters.

The provision for weapons in the model is as follows: As many as four weapon types, with their associated kill probabilities as a function of range to the helicopters, can be assigned to the armored unit. The attack helicopters are armed with missile type weapons which can have either nuclear warheads with an associated CEP as a function of target range and crossing speed or conventional warheads with an associated kill probability.

Assigned to each participant in the model is a probability of seeing opposing participants as a function of range given that the opponents are in view of one another. The model begins the conflict between the opponents by proceeding to move all forces according to plan until an opponent is sighted by one of the participants. At this time a decision to engage the sighted target is considered by the participant making the sighting. If the decision is not to engage, the computer model is time stepped until an engagement decision is reached or the game run is completed. If, however, the decision is to engage the sighted target, the attack phase of ARMREC is executed. This phase of the program is described in Section 5.

For location purposes all three participants are considered as point forces, i.e., all elements making up the forces of any of the three combatants are considered to occupy the same point, (X,Y,Z), at any given time in the simulation. The routines that govern the movement of the helicopter sections and armored unit until the attack phase is executed are described in Sections 3 and 4.

At the conclusion of each attack phase or engagement, the ARMREC program is designed to collect and print results which are necessary for assessing the impact of parameter changes which can be made in the program. A complete description of both the inputs to the program and the printed results is found in APPENDIX III.

The scenario used with the ARMREC program consisted of an attack section and a scout section each containing two helicopters. The armored mobile land unit consisted of 3 tanks and 2 armored personnel carriers. The flight path of the helicopters was both preplanned and NOE generated by the program. The time step for the model was set at one second.

3. Helicopter and armored unit movement

The tracks of the two helicopter sections and the armored unit on the terrain are input values to the ARMREC program. The terrain heights of points along each of the three tracks are computed using the appropriate approximating polynomial expression generated by the program TERRAIN. These heights are used to generate a nap of earth flight path when desired. A nap of earth, NOE, flight path is defined as flying as close to the ground as vegetation and obstacles permit, taking full advantage of the terrain to gain concealment and cover from the enemy's observation and fire (8).

Helicopters proceed along their preplanned flight path and a check is made during every program time step to determine if they can see the land target. If it is determined that no line of sight has existed between the helicopters and the land target for a specified period of time, the scout helicopters perform a "pop-up" maneuver. This pop-up tactic is described as suddenly ascending vertically from behind terrain features in order to see any enemy ground units that might be present. This tactic, combined with NOE flight, has been found to improve the chance of successful mission completion by scout and attack helicopters against a variety of targets (1).

A helicopter while in the pop-up phase, is allowed to climb to a predetermined height or until interacquisition is established. If it is determined that no visual detection has occurred or is possible at the end of the pop-up ascent, the helicopter is programmed to continue its NOE flight path.

4. Interacquisition

Inherent to the ARMREC program is the assumption that whenever an unobstructed line of sight exists between two participants, they both have the possibility of visual detection.

To determine if two participants can possibly see one another a line, called the line of sight, is projected between them. If the elevation of any point on the line of sight projected onto the terrain surface is greater than the corresponding point on the line of sight, the participants are considered not in view of one another. If all points on the line of sight are higher than the elevations of their respective projected points on the terrain surface, then the line of sight is unobstructed and visual detection can occur. This computation is carried out in the ARMREC program by checking points 50 meters apart, on the average, on the line of sight.

If visual detection can occur, the detection probability as a function of range associated with each participant is then compared with a generated random number to ascertain if contact has been made by one or both of the participants. If it is determined that a visual contact exists, the ARMREC program executes the attack phase. If contact is not established or the participants are not in view of one another, then the participants repeat the cycle of events in the next time step.

5. Attack

The attack phase of ARMREC provides the logic for the participants during the engagement portion of the simulation. As mentioned in the previous section, the attack phase commences when either of the opposing participants visually detects the other. The principal logical divisions for this phase are as follows:

Armored Unit Movement

The armored land unit proceeds on the basis of three assumptions. These assumptions are: The unit follows a predetermined route to a selected destination with no deviation from the route allowed; the unit must stop to fire; and elements of the unit will fire whenever a helicopter is in view and ammunition is available.

Aircraft Initial Decisions

When one of the helicopter sections has sighted a target, a firing point is computed for the attack helicopter on a constant bearing from the target to the attack section at a given attack range, ATR, . A scout observation position is then chosen along a line perpendicular to the attack-target axis and at a distance prescribed by ATR.

If the scout section can obtain an unobstructed view of both the target and the attack section's computed firing point, a refinement of the firing position is made. This refinement consists of selecting a point, if one exists, in the vicinity of the initial computed firing point such that an uninterrupted line of sight, LOS, exists to the target. If no such point exists a refinement is not made. A major assumption in the computation of these positions is that the attack group verses armored unit speed ratio ranges from 2.5 to 3.0.

Position Determination

During each time step the positions of the participants are calculated and the lines of sight between the attack and scout sections and the armored unit are evaluated to determine whether or not they are unobstructed. If the line of sight between the attack section and the armored unit is discontinuous as the attack section approaches the firing point, the scout section altitude is modified to simulate a pop-up tactic. This tactic is utilized to insure that the scout section will be able to assist the attack section in target position orientation.

Armored Unit Gunfire Control

Subsequent to the calculation of positions and the lines of sight evaluation, it is determined if the land unit could or would have fired at the helicopters during the previous time step. The ability of each weapon type to fire is predicated upon range, availability of ammunition, whether sufficient time has elapsed to acquire the helicopter, i.e., time to train, sight and prepare the weapon for firing, and when applicable, whether enough time has passed to simulate reloading and re-acquiring a helicopter.

If the armored unit is unable to fire at one or both of the helicopter sections due to the loss of a previously established visual contact, a decision is made for the unit to remain in place or to proceed to its destination at an emergency speed, TSPDM. This speed is maintained until at least one of the helicopter sections is reacquired at which time the armored unit's movement is determined as previously described, or until its destination is reached. The decision to remain or proceed at emergency speed to the destination is

based upon the length of elapsed time before the lost contact can again be seen. This is programmed to account for the momentary obscuration of a helicopter section by a terrain feature. Time intervals to re-acquisition of less than 15 seconds were arbitaraily selected as a basis for the armored unit to maintain a fixed position.

The armored unit will fire at a helicopter if the probability of a kill as calculated by the following equation, is at least .15 .

$$P(\text{helicopter kill}) = 1-\prod_{i=1}^{NW} (1-P_i(r)^{NWi})$$

Where:

NW = The number of weapon types

NWi= The number of weapons of type i

$P_i(r)$= The probability of helicopter kill by weapon type i at range r .

In the event the armored unit is able to fire at both helicopter sections a predetermined firing doctrine, IKILL, is used in the model to decide which target(s) the armored unit will fire at. The firing doctrine gives firing priorities for the helicopter section type, i.e., scout or attack, or it gives priority to the section with the lowest probability of survival. The armored unit's weapons can also be apportioned, by weapon type, to the helicopter section against which it has the highest probability of a kill.

Aircraft Fire Control

In general, the attack helicopters are able to fire under the following conditions: the armored unit is in visual contact and will remain so for the time of flight of the helicopter's missile; a period of time, HTF, simulating acquisition, has passed since visual

contact was made, and missiles are available for firing. Obscuration of the target due to missile detonation does not affect the firing capabilities of the attack section. In particular, a nuclear missile can be fired at ranges in excess of 500 meters, and a conventional rocket can be fired at ranges in excess of 100 meters from the target. If the above conditions are not fulfilled or a helicopter has fired a missile which has not yet impacted then the helicopter may not fire. Once a helicopter has fired it must maintain the course and speed it had at the time of fire until missile impact. Further firing or evasive maneuver is then permitted. If the helicopter cannot fire for reasons other than just having fired, evasive maneuver or movement to press the attack is allowable. In either case, if missiles are available, once evasive maneuver has been taken, another firing point is selected and routes of flight are computed as described before. If all missiles have been expended the program generates a route from the attack section's position to a point of concealment and then to a terminal point.

The scout section's route during the gunfire control phase of the program is also subject to evaluation. Once the scout section establishes visual contact with the armored unit a route is selected to allow the scout to shuttle between an observation point and a point concealed from the land unit's view. This continues until the attack section is destroyed or has expended its missiles, or until the land unit is destroyed. ARMREC then calculates a flight path from the scout section's present position, via a concealed point to its terminal position.

A major assumption is that land unit weapons are considered effective in the time step in which they are fired while the missiles fired from helicopters have a time of flight extending over multiple time steps.

6. Destruction Assessment

A helicopter is considered to be out of action if a number drawn from a uniform distribution is less than or equal to the probability of a kill for the weapons employed against the aircraft. No more than one helicopter in each section may be destroyed in a time step.

As mentioned earlier, two missile types can be delivered by an attack helicopter, one a conventional and the other a nuclear. The conventional missile is merely delivered with a given kill probability. That is, for conventional weapons the decision as to whether an element of the armored unit is killed or not is a simple yes or no determined by comparing the probability of kill with a generated random number. In this sense the damage inflicted on an element of the armored unit is either none or total and this must be reflected in the numerical values assigned to the probability of kill. When a nuclear weapon is delivered a CEP is generated in order to fit a normal distribution to the distribution of points of impact about the aim point. The CEP is generated as a linear function of range to the target in meters and the relative crossing speed in meters per second. The functional relationship of range and relative crossing speed to CEP used in ARMREC was initially established to produce an average direct hit probability of near .40 at target attack ranges of 900 to 1400 meters. However, with minor program modification, known CEP's for weapons delivery systems can be used. The missile aimpoint is determined as a function of target speed and projected direction of motion of the target.

The relative position of the nuclear projectile impact point to the center and end points of the armored unit, together with the target radius of destruction for the given weapon yield , provides the

information necessary to determine the extent of damage inflicted on the target. An element, including its assigned weapon types, is considered out of action if it lies within the target radius of destruction projected from the impact point of the missile.

Only surface bursts of small yield nuclear missiles are considered in the ARMREC program. With the low yield projectile envisioned as being used, it is felt that no detrimental nuclear effects would be experienced by the helicopter weapons system at the attack ranges considered. However, provision is made in the ARMREC program to determine the overpressure produced by the nuclear burst at various attack ranges since overpressure is found to be the critical damage factor on aircraft weapon systems (6). First order mach effects were considered the main augmenters in calculating the overpressure. The function which estimates the overpressure from a nuclear blast of given yield at varied ranges can be found in D.A.S.A. publications (9).

It is also assumed in the computer model that the area within the target complex is flat earth. This may affect the damage assessment since the armored unit critical damage factor in determining the target radius of destruction was the nuclear radiation intensity inflicted upon the personnel within the target complex (3).

It is also assumed in the model that armored unit destruction and aircraft damage is simultaneously known to all participants.

7. Comments

In addition to comments pertaining to assumptions made earlier, this section contains remarks that are considered significant.

All participants described in the model moved on or above the terrain without regard to natural vegetation or man made obstacles. No attempt was made in the model to degrade sighting probabilities as a result of the masking or camouflaging effects that are produced by these elements. Cover afforded by vegetation could be simulated by allowing ground elements to operate below the terrain surface in areas where masking is prevalent. Area extent would be controlled to approximate light or heavy coverage. Background concealing effects by terrain or vegetation on helicopter acquisition by ground targets could be simulated by sensing terrain along the extended line of sight between ground and airborne elements. A diminishing factor could then be applied to the probability of aircraft acquisition.

In the nuclear effects assessment portion of the program it was assumed that the vehicles of the armored unit were spaced 50 meters apart and in single file. Modification of the damage part of the program would be necessary in order to calculate the damage inflicted upon other target types.

It is also the case that the model scenario has been described in terms of two sections of helicopters and an armored unit. Actually the structure of the model is such that other participant types may be used in place of those described by changing the input parameters which determine the operating envelopes of the participants. As a result many other conflict type scenarios may be investigated by this model. Fixed ground participants, such as those used in Capt. J. L. Harrison's thesis (2), may be incorporated into ARMREC with a mini-

mum of programming.

The previous sections have described the model in terms of the logical development of actions and reactions. It has not been specifically mentioned that all of the previous sections describe parts of an integrated complex computer program. The ARMREC program is composed of a main program and eleven sub-routines. The main program initiates each run, directs the movement of the participants to the attack phase, and directs the execution of the historical print out tableau at the completion of each run, Fig. 3. The names and functions of the sub-routines are as follows:

BNDCK checks for out of bounds conditions as routes are computed for each of the moving participants. If a route is found to extend beyond the map area upon which the duel is designed to take place, BNDCK initiates a program stop.

RANGE is used to calculate the position of each participant on its computed path at any time step, and senses when a participant reaches its terminal position.

ALOS calculates the range between two positions and determines if an unobstructed line of sight exists between them.

ELEV uses the input coefficients of the terrain approximating polynomials to calculate the elevation at any surface point (X,Y) within the game boundaries.

CONTOUR computes a nap of the earth flight path between points along the track of an aircraft.

TGSE selects a position such that a LOS exists, or one in which a LOS does not exist between the position and another specified point.

RTE determines the X,Y coordinates of a path such that nap of the earth flight is possible between any two specified points. It enables an aircraft to simulate a spiraling climb to elevations which normal nap of the earth flight as simulated in CONTOUR does not allow.

PLOT1 provides the necessary data for the graph output desired.

ATTACK provides for the functions mentioned in Section five. It also calculates the aircraft damage due to land unit fire as described in Section six.

PRINT1 controls the printed historical output.

NUC determines damage inflicted upon participants by nuclear and conventional missiles as described in Section six.

DRAW, which does not appear in the program listing, produces the output on tape necessary for the plotting of graphs. A brief description of DRAW is included within APPENDIX IV.

A complete listing of program ARMREC and a complete set of flow charts presenting the initial program logic are included as APPENDIX II and I respectively. A description of all computer input and output data appears in APPENDIX III.

8. Conclusions.

The real value of this simulation, as felt by its authors, is the insight gained while trying to represent the important ingredients in the simple duel developed in this thesis. An awareness of the pitfalls and difficulties inherent in a simulation and the necessary compromises with reality to obtain manageable scope within a limited time are important and invaluable aids in evaluating the results of this and other simulations. The modeling or simulation of an event, even of comparatively little magnitude, is the genesis of an almost unending sequence of investigations into all of its ramifications.

In particular, the authors feel that more sophistication is required before the model can be called complete. Airborne and ground element capabilities must be extended to more nearly approximate their operating envelopes. Several variables treated as deterministic in the model should be considered as probabilistic, e.g., all type weapons fire as a unit and pop-up's occur at definite time intervals. It should also be noted that data used to generate the program results is not assumed to be accurate. Kill probabilities and sighting probabilities, for example, are but very rough estimates assumed by the authors.

Time was not available to use program ARMREC in making a detail analysis of the results of the interactions between the opposing participants by changing the parameters and tactics in the thesis scenario. However, some general observations can be made. Excluding the effects of vegetation and man made obstacles, NOE flight with a pop-up tactic seemed to enhance the helicopters' capability to detect and successfully engage the land unit. Based on missile speeds of

350-370 meters per second firing at ranges of 1000-1500 meters and an acquisition time required by the helicopter in excess of three to four seconds, program results also indicates the requirement that a helicopter have a continuous LOS during missile flight significantly lowers the probability of helicopter survival because of the time available for the firing of land unit weapons.

BIBLIOGRAPHY

1. Hoppes, H.N. An evaluation of helicopter pop-up tactics. Research Analysis Corporation, RAC-T- 464, November 1965.

2. Harrison, J.L. A computer simulation of an aircraft penetration over hostile terrain, Thesis U. S. Naval Postgraduate School, May 1965.

3. Ethridge, N.H. Blast effects on simple objects and military vehicles (Secret), Operation Sunbeam, Little Feller, Ballistic Research Laboratories, POR-2261, September 1964.

4. Hoppes, H.N. Reconnaissance techniques for light observation helicopters in a summer environment - A two sided field play, Research Analysis Corporation, RAC-T-433, August 1964.

5. Staff officers field manual - nuclear weapons employment, FM 101-31-2 (Secret) and FM 101-31-3, Department of the Army, February 1963.

6. Glasstone, S. The effects of nuclear weapons, U.S. Defense Atomic Support Agency, April 1962.

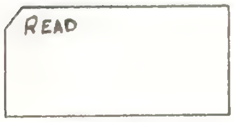
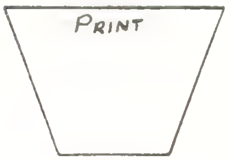
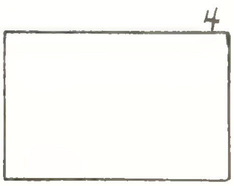
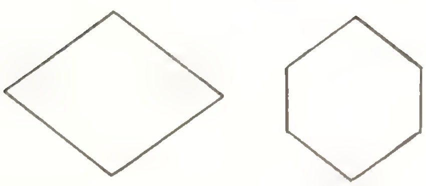
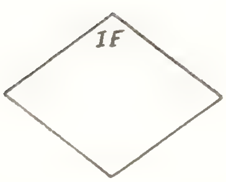
7. Von Senger Und Etterlin, F.M. The worlds armoured fighting vehicles, Doubleday and Co., 1962.

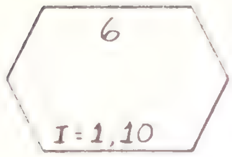
8. Armoured cavalry platoon and troop, air cavalry troop, and division armored cavalry squadron, Department of the Army, December 1961.

9. Moulton, J.F., Jr. Nuclear weapons blast phenomena, Volumes I and II (Secret), Defense Atomic Support Agency, March 1960.

APPENDIX I

This appendix contains the flow charts of the ARMREC computer program logic. The following conventions will be used within the chart.

Symbol	Symbol definition
	Read the input data listed
	PRINT statement
	A FORTRAN statement, the number 4 on the upper right hand corner is the statement number. If the rectangle contains a call statement the page on which the subroutine starts, immediately follows the subroutine name, e.g., P-4 means flow for subroutine also starts on page 4. Rectangle may be of any size.
	LOGICAL IF statement.
	COMPUTED "GO TO" statement.
	IF statement.



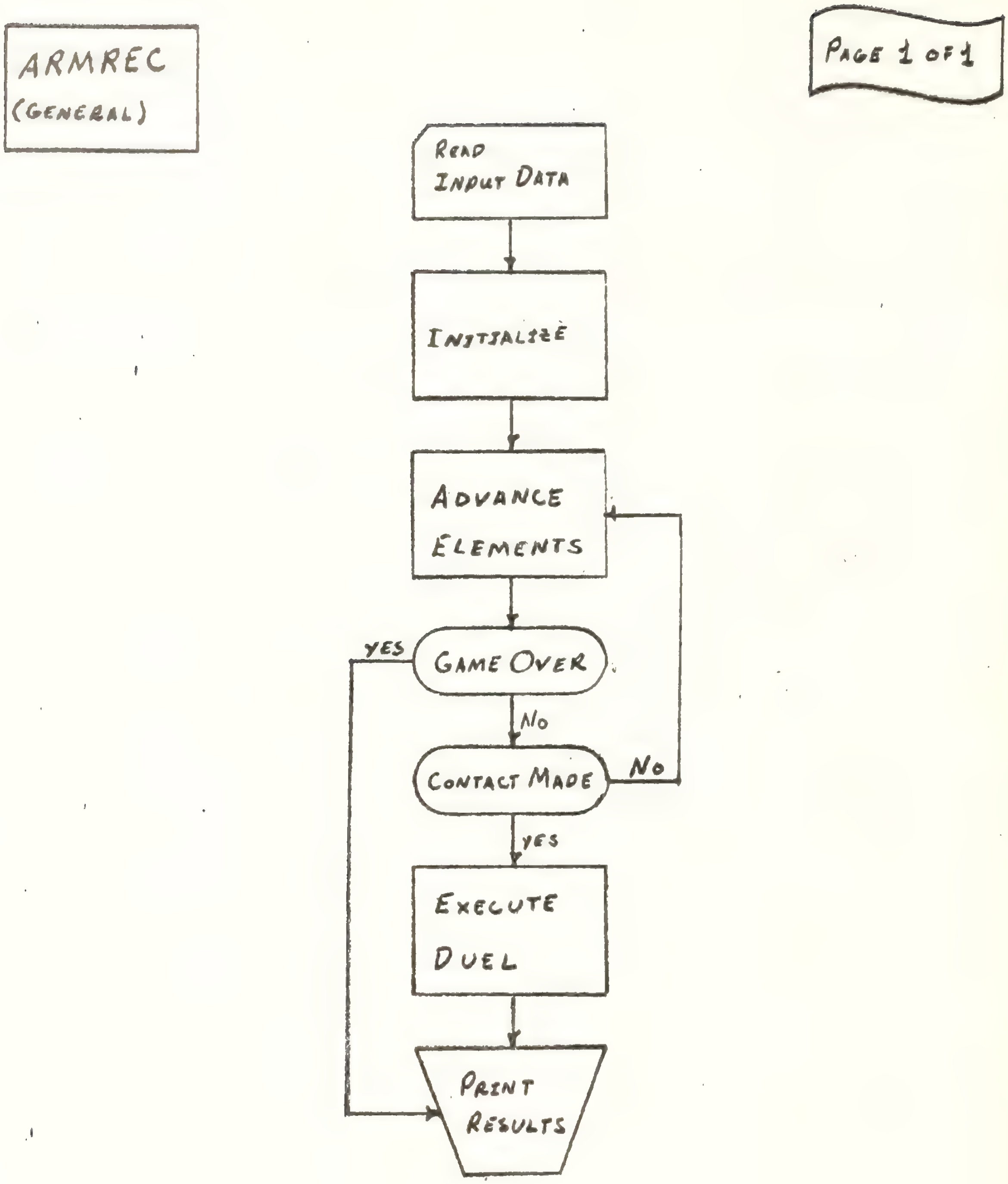
DO statement
,eg., "Do 6 I=1,10"

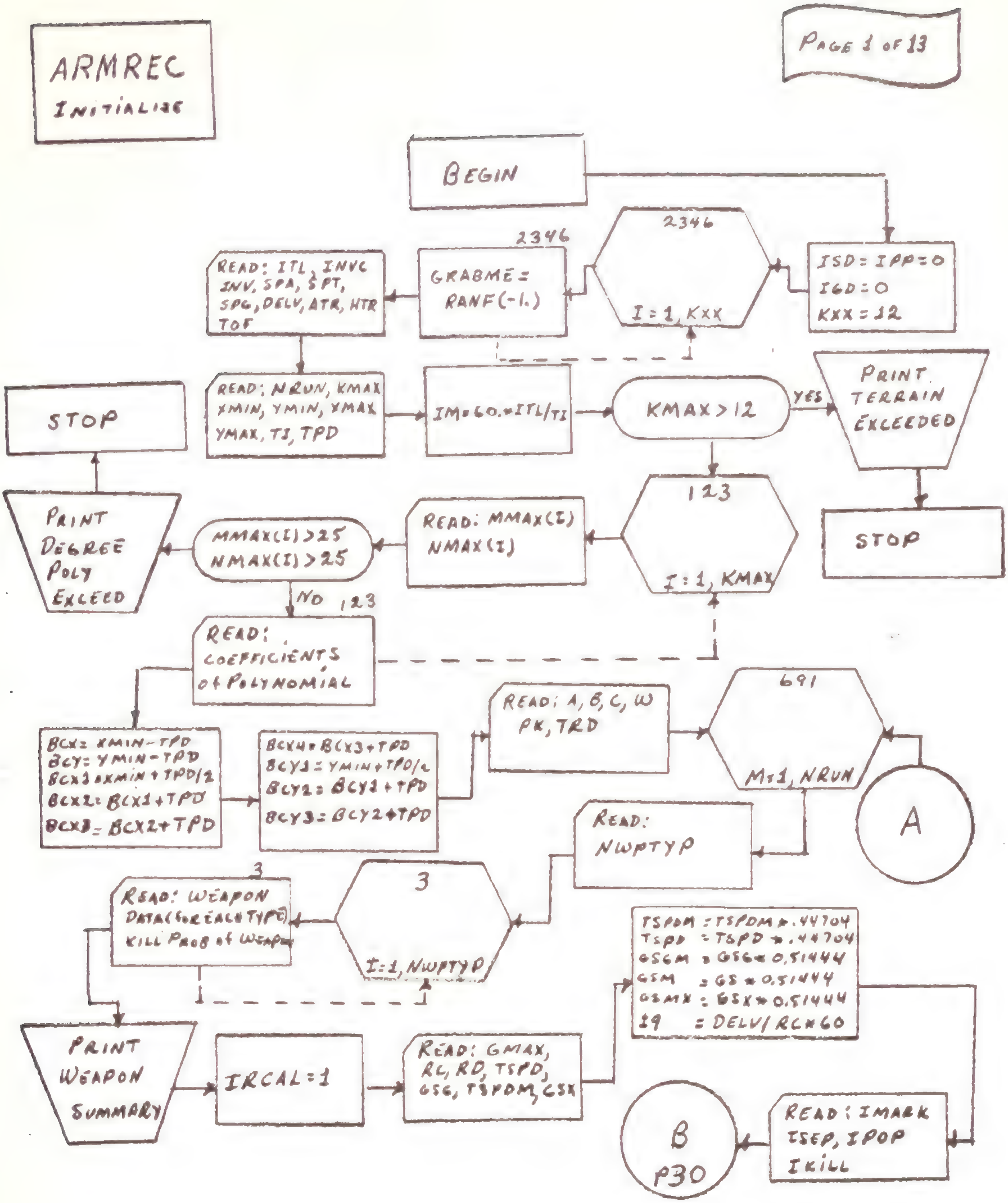
Used for BEGIN,END,RETURN, or
STOP statements

Connector between statements in the same program or subroutine which appear on different pages or cannot be connected by an unbroken line.

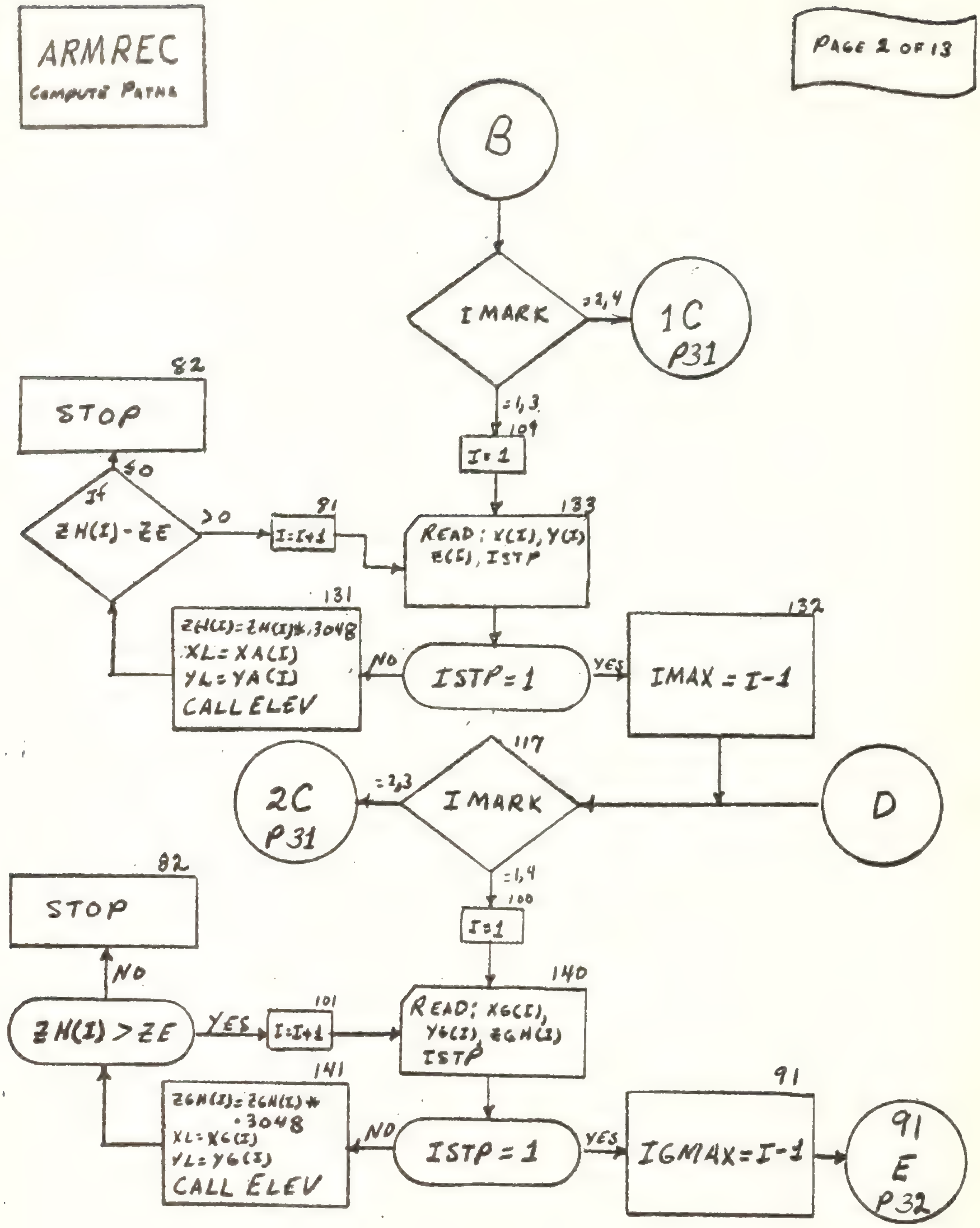
A= connector symbol
163= statement number to which it connects if there is one
P70= connector appears on page 70.

CONTINUE statement


ARMREC
(GENERAL)
PAGE 1 OF 1
READ INPUT DATA
INITIALIZE
ADVANCE ELEMENTS
GAME OVER
YES
NO
CONTACT MADE
NO
YES
EXECUTE DUEL
PRINT RESULTS

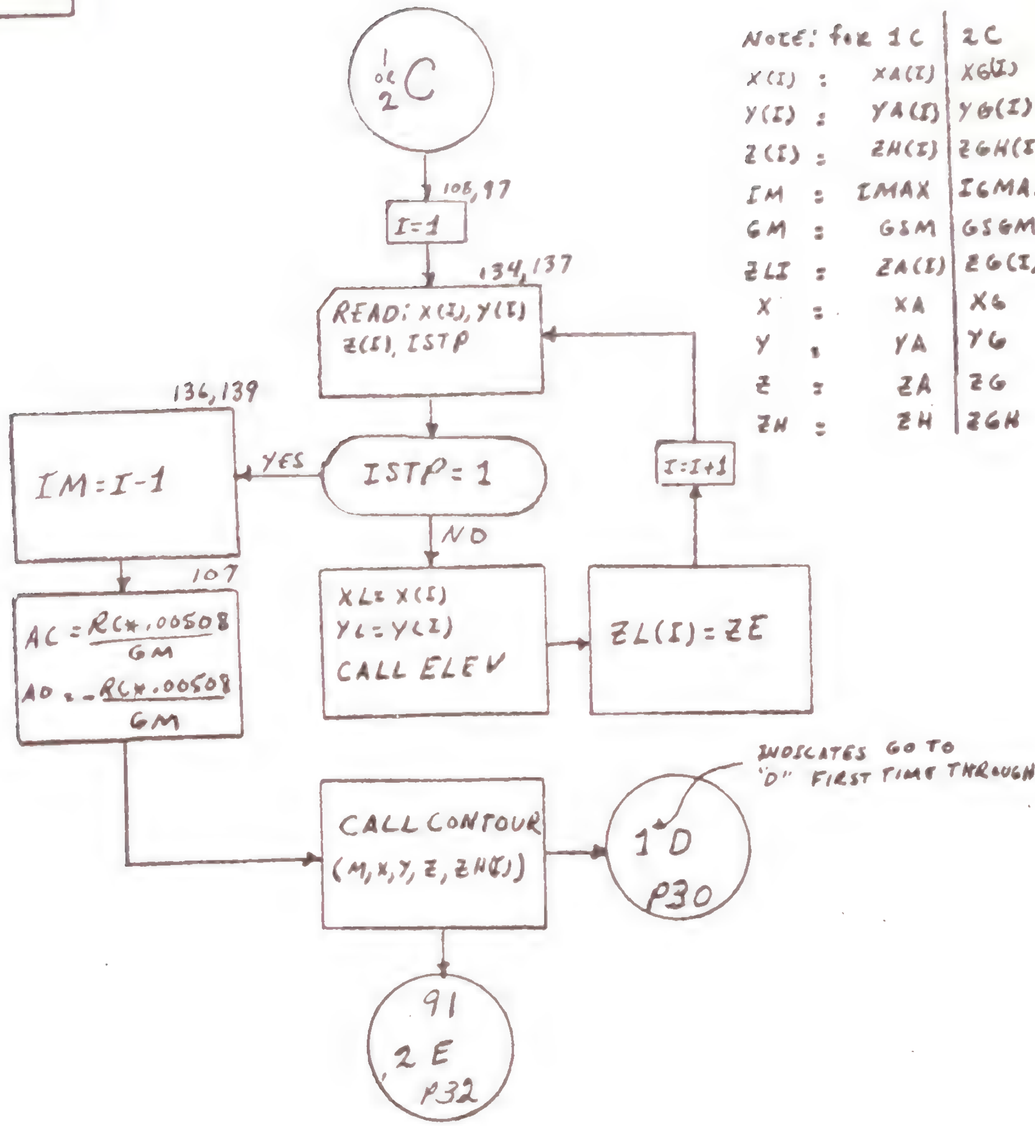


ARMREC
INITIALIZE
PAGE 1 OF 13
BEGIN
2346
ISD = IPP = 0
IGD = 0
KXX = 12
I = 1, KXX
GRABME = RANF(-1.)
READ: ITL, INVC, INV, SPA, SPT, SPG, DELV, ATR, HTR, TOF
READ: NRUN, KMAX, XMIN, YMIN, XMAX, YMAX, TI, TPD
IM = 60.*ITL/TI
KMAX > 12
YES
PRINT TERRAIN EXCEEDED
STOP
123
I = 1, KMAX
READ: MMAX(I) NMAX(I)
MMAX(I) > 25 NMAX(I) > 25
PRINT DEGREE POLY EXCEED
STOP
NO
READ: COEFFICIENTS OF POLYNOMIAL
BCX = XMIN-TPD
BCY = YMIN-TPD
BCX1 = XMIN + TPD/2
BCX2 = BCX1 + TPD
BCX3 = BCX2 + TPD
BCX4 = BCX3 + TPD
BCY1 = YMIN + TPD/2
BCY2 = BCY1 + TPD
BCY3 = BCY2 + TPD
READ: A, B, C, W, PK, TRD
691
M = 1, NRUN
A
READ: NWPTYP
3
I = 1, NWPTYP
READ: WEAPON DATA (FOR EACH TYPE) KILL PROB OF WEAPON
PRINT WEAPON SUMMARY
IRCAL = 1
READ: GMAX, RC, RD, TSPD, GSG, TSPDM, GSX
TSPDM = TSPDM * .44704
TSPD = TSPD * .44704
GSGM = GSG * 0.51444
GSM = GS * 0.51444
GSMX = GSX * 0.51444
I9 = DELV/RC*60
READ: IMARK ISEP, IPOP IKILL
B
P30

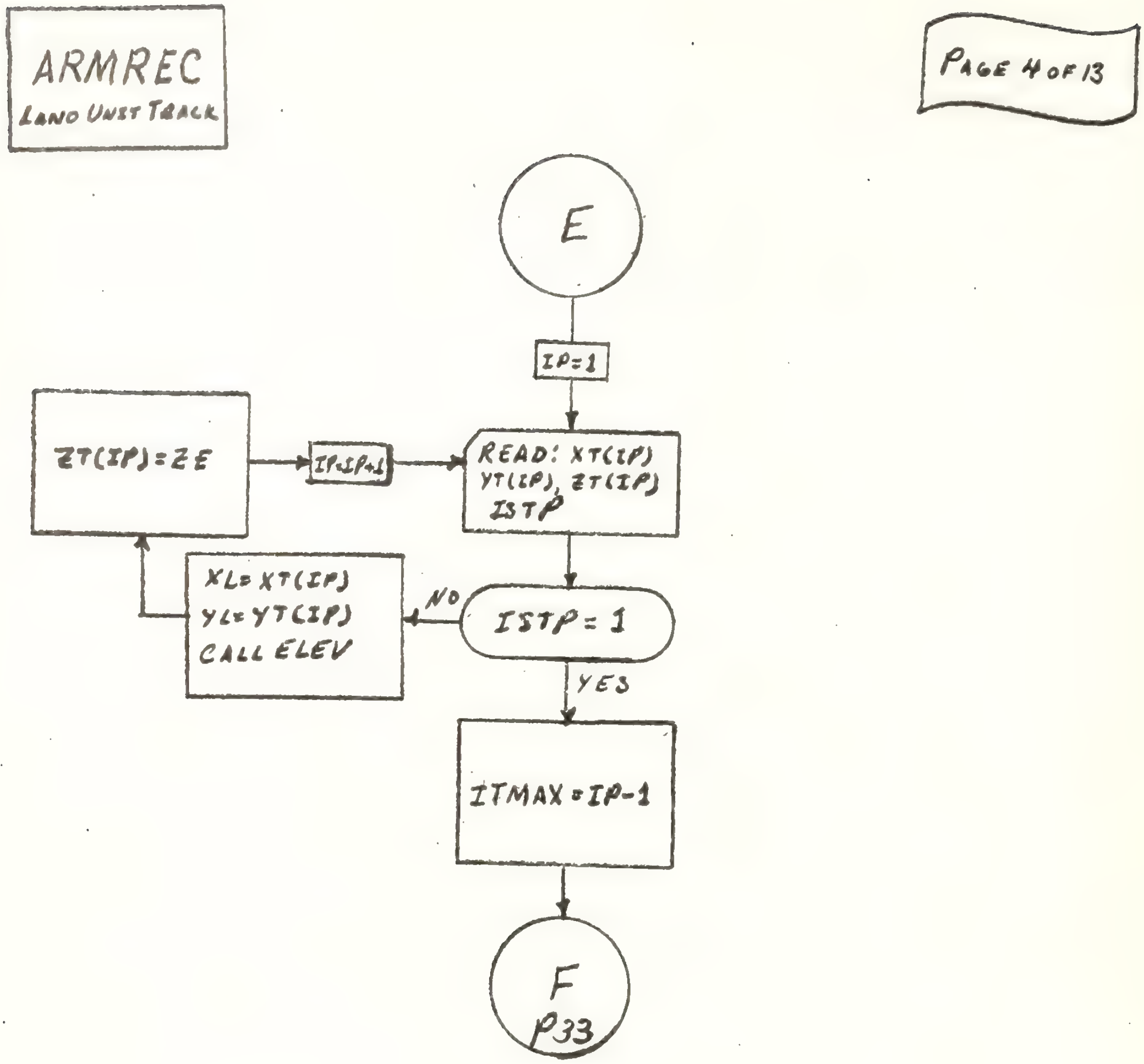

ARMREC
COMPUTE PATHS
PAGE 2 OF 13
B
IMARK
=2,4
1C
P31
=1,3
109
I=1
82
STOP
80
If
ZH(I) - ZE
>0
91
I=I+1
133
READ: X(I), Y(I)
Z(I), ISTP
131
ZH(I)=ZH(I)*.3048
XL=XA(I)
YL=YA(I)
CALL ELEV
NO
ISTP=1
YES
132
IMAX = I-1
117
2C
P31
=2,3
IMARK
D
=1,4
100
I=1
82
STOP
NO
ZH(I) > ZE
YES
101
I=I+1
140
READ: XG(I),
YG(I), ZGH(I)
ISTP
141
ZGH(I)=ZGH(I)*
.3048
XL=XG(I)
YL=YG(I)
CALL ELEV
NO
ISTP=1
YES
91
IGMAX=I-1
91
E
P32

ARMREC

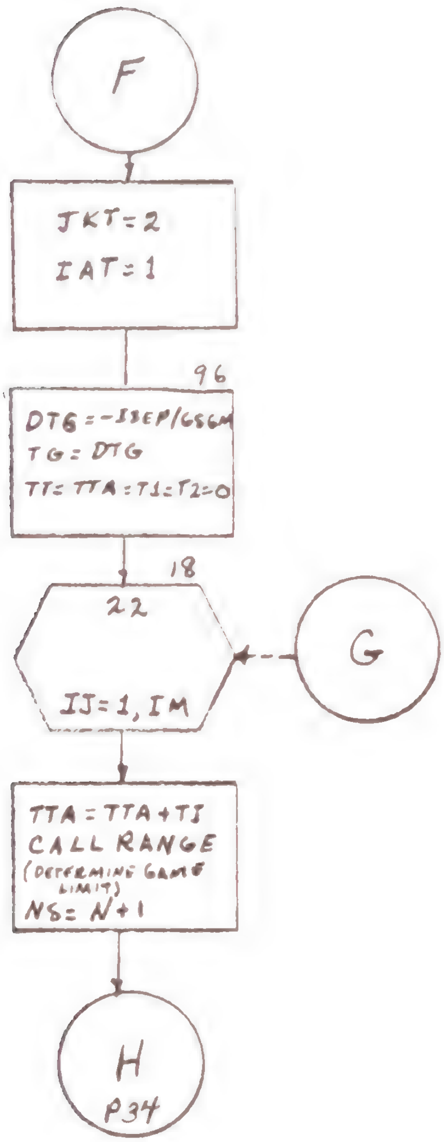
NOE FLIGHTPATH



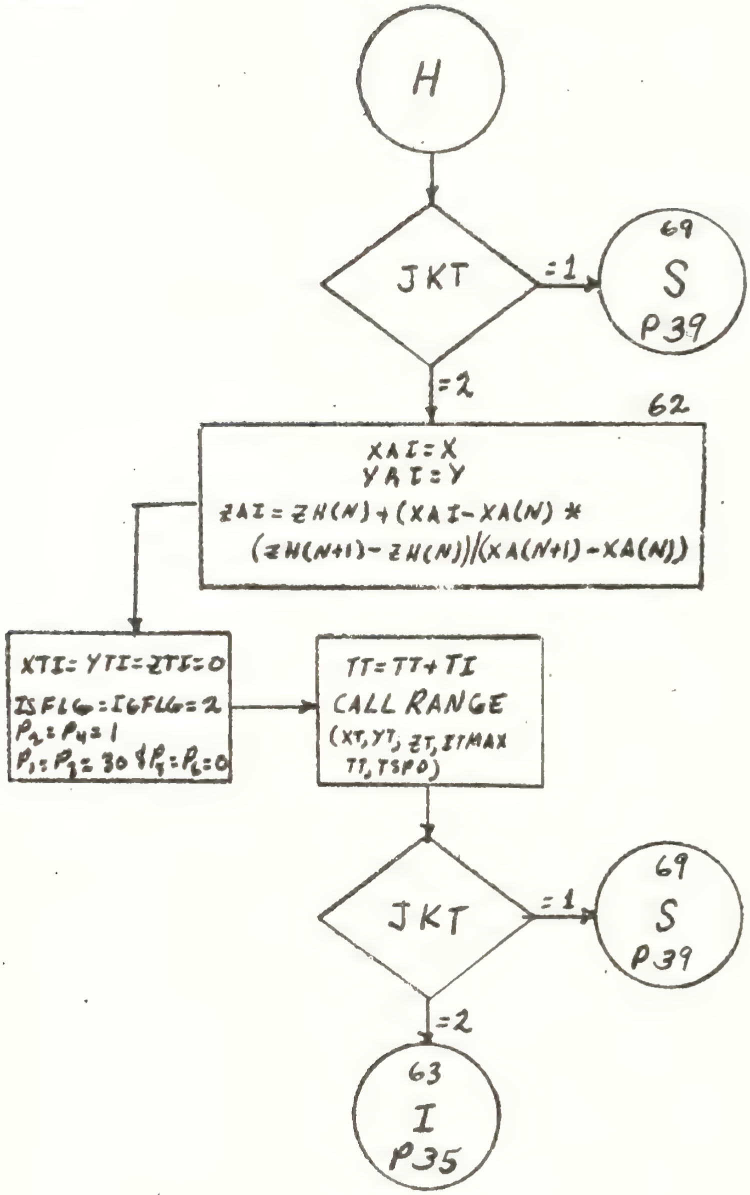
NOTE: for		1C	2C
X(I)	:	XA(I)	XG(I)
Y(I)	:	YA(I)	YG(I)
Z(I)	:	ZH(I)	ZGH(I)
IM	:	IMAX	IGMAX
GM	:	GSM	GSGM
ZLI	:	ZA(I)	ZG(I)
X	:	XA	XG
Y	:	YA	YG
Z	:	ZA	ZG
ZH	:	ZH	ZGH


ARMREC
LAND UNIT TRACK
PAGE 4 OF 13
E
IP=1
READ: XT(IP)
YT(IP), ZT(IP)
ISTP
ISTP = 1
NO
YES
XL=XT(IP)
YL=YT(IP)
CALL ELEV
ZT(IP)=ZE
IP=IP+1
ITMAX=IP-1
F
P33

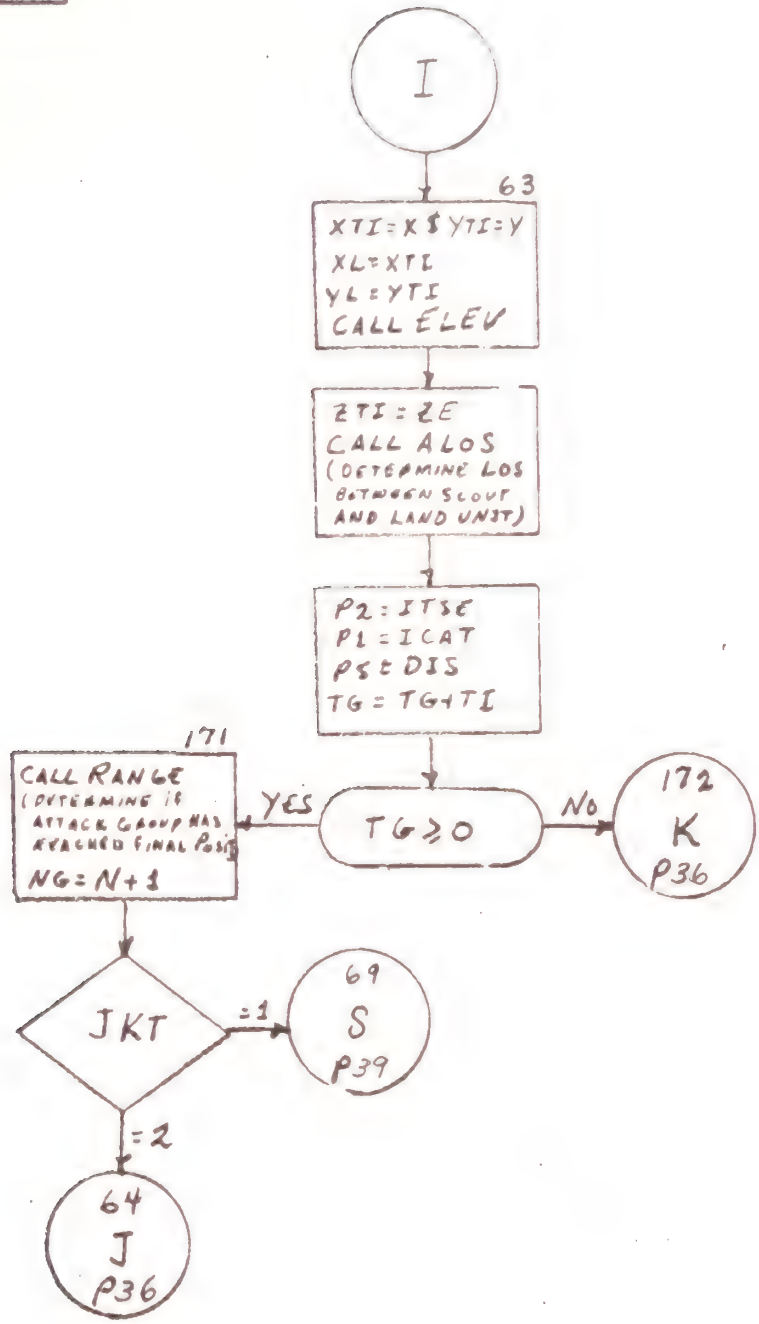
ARMREC
MOVEMENT TO CONTACT

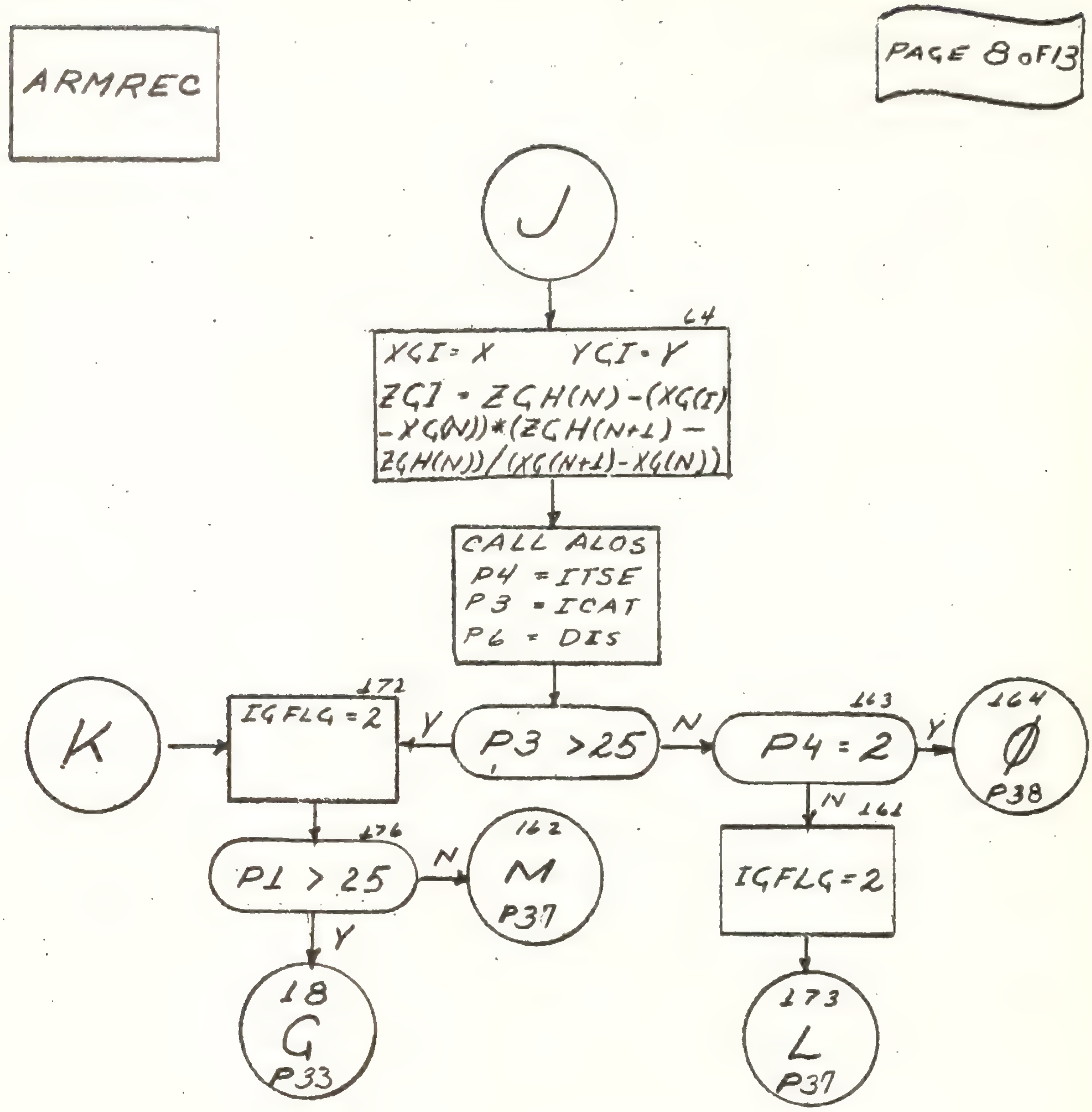


ARMREC
MOVEMENT TO CONTACT

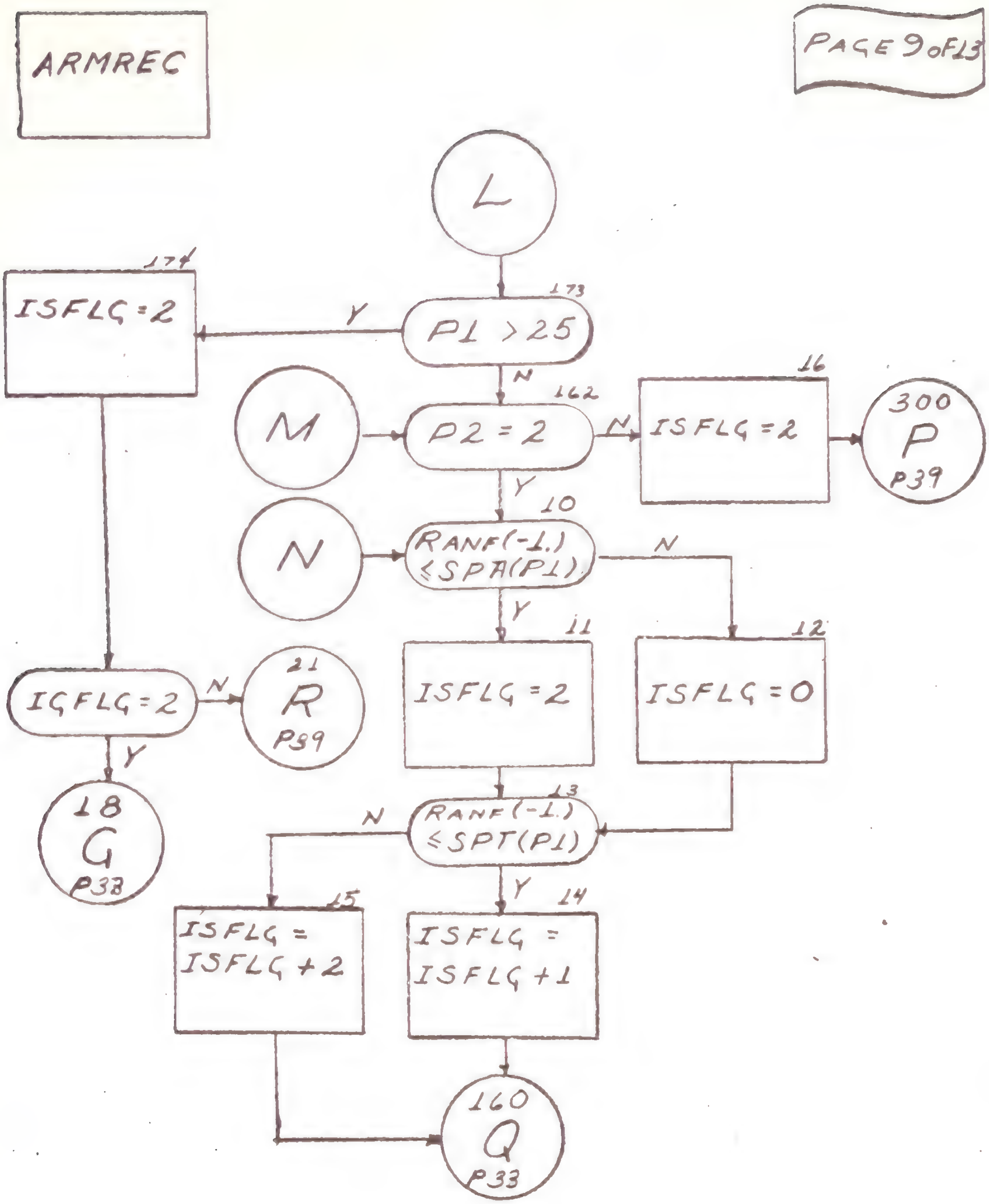


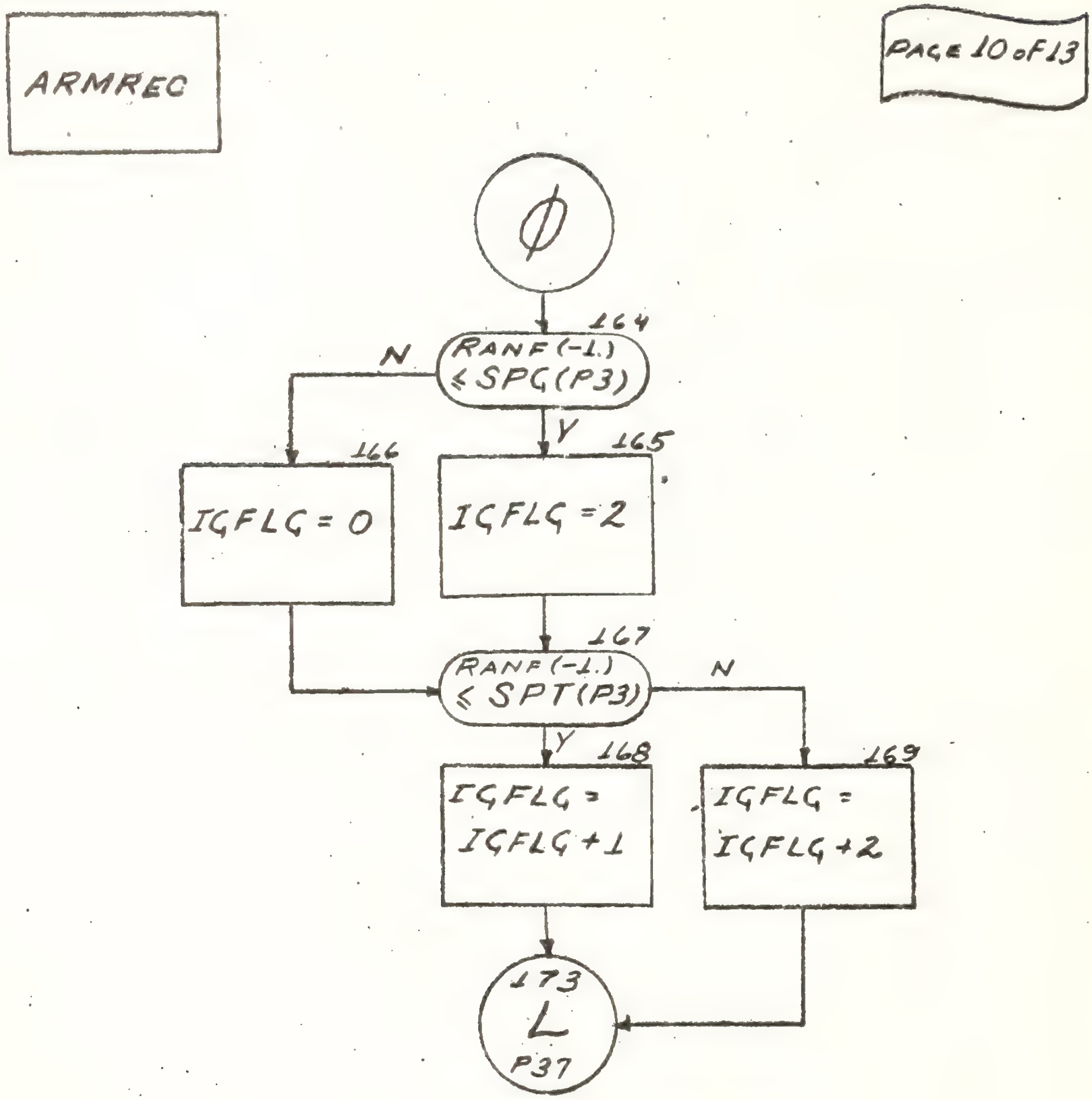
ARMREC


I
63
XTI=X $ YTI=Y
XL=XTI
YL=YTI
CALL ELEV
ZTI=ZE
CALL ALOS
(DETERMINE LOS
BETWEEN SCOUT
AND LAND UNIT)
P2=ITSE
P1=ICAT
PS=DIS
TG=TG+TI
TG≥0
YES
NO
172
K
P36
171
CALL RANGE
(DETERMINE IF
ATTACK GROUP HAS
REACHED FINAL POSIT
NG=N+1
JKT
=1
69
S
P39
=2
64
J
P36

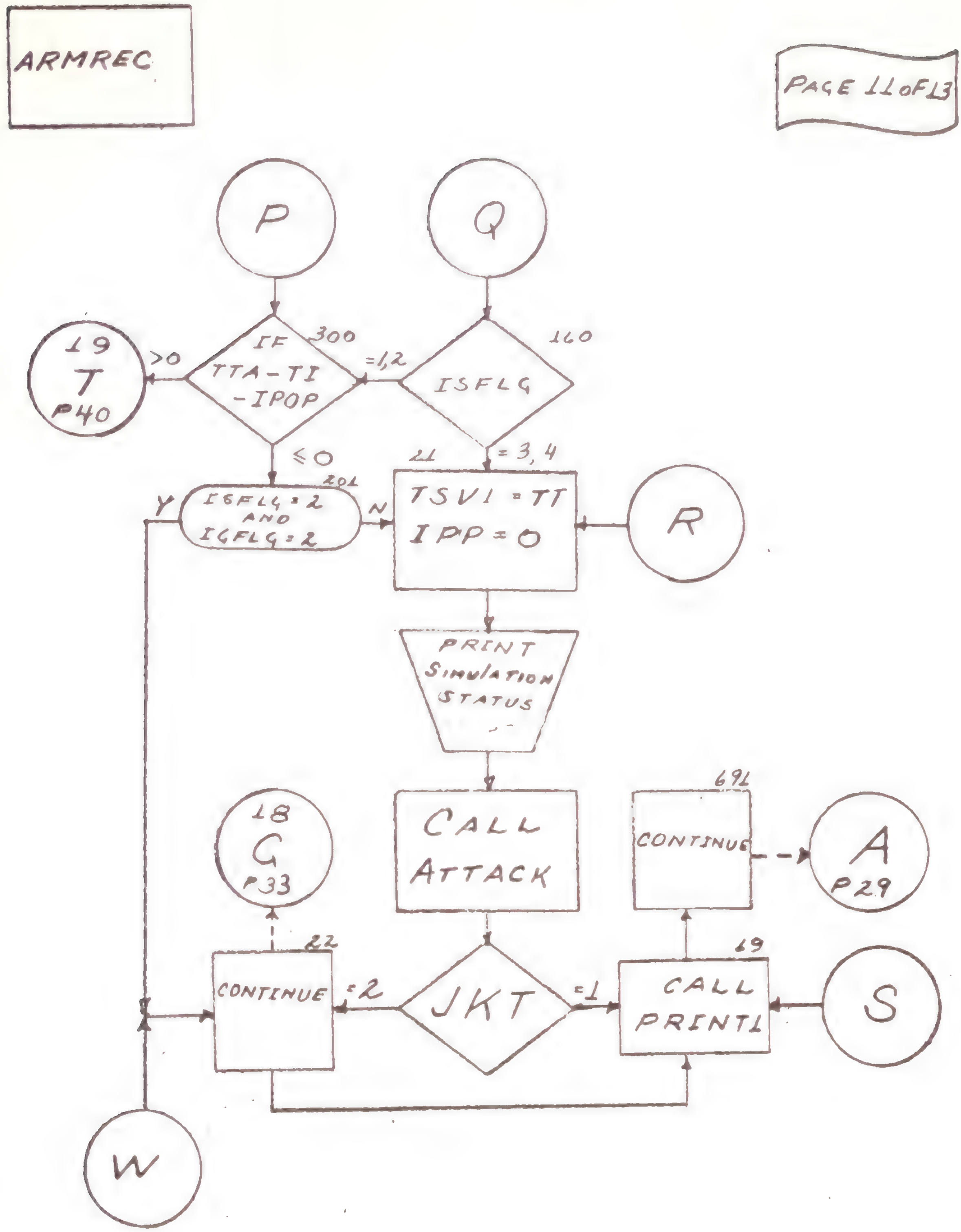


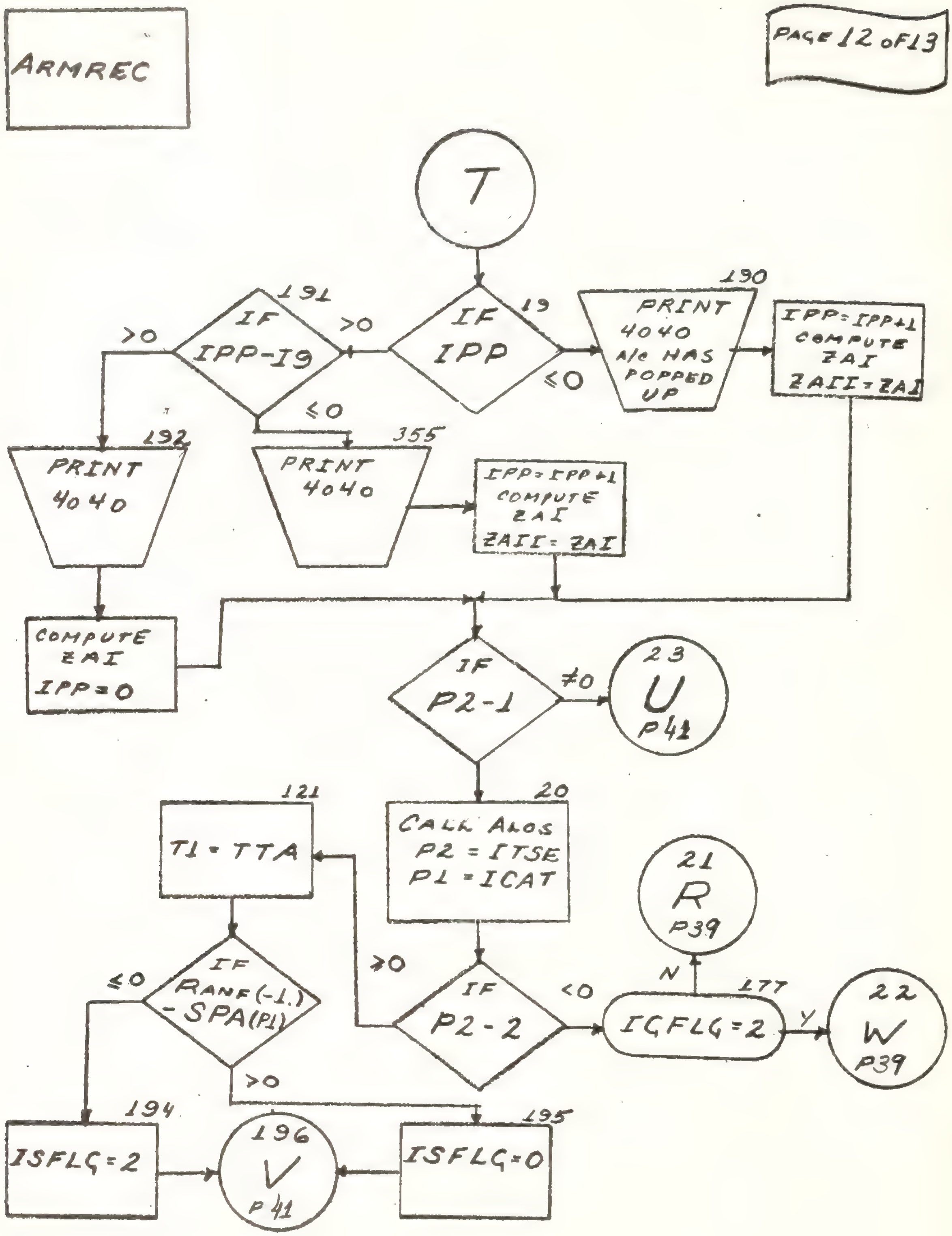
ARMREC
PAGE 8 OF 13
J
64
XGI = X YGI = Y
ZGI = ZGH(N) - (XG(I) - XG(N)) * (ZGH(N+1) - ZGH(N)) / (XG(N+1) - XG(N))
CALL ALOS
P4 = ITSE
P3 = ICAT
P6 = DIS
K
172
IGFLG = 2
Y
P3 > 25
N
163
P4 = 2
Y
164
Ø
P38
N
161
IGFLG = 2
176
P1 > 25
N
162
M
P37
Y
18
G
P33
173
L
P37

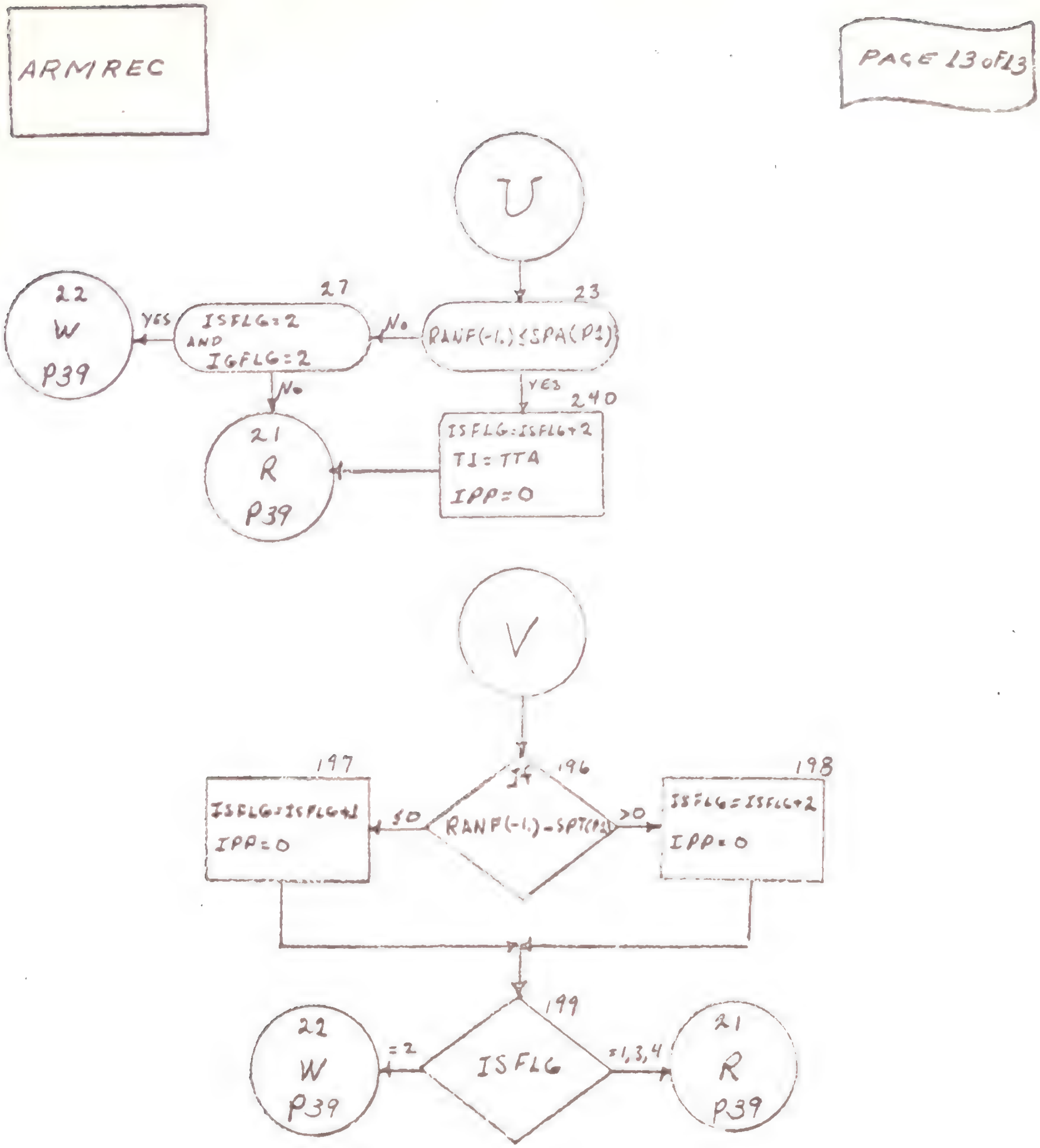

ARMREC
PAGE 9 OF 13
L
173
P1 > 25
Y
N
174
ISFLG = 2
M
162
P2 = 2
N
16
ISFLG = 2
300
P
P39
Y
N
10
RANF(-1.)
≤ SPA(P1)
N
Y
11
ISFLG = 2
12
ISFLG = 0
IGFLG = 2
N
21
R
P39
Y
18
G
P38
13
RANF(-1.)
≤ SPT(P1)
N
Y
15
ISFLG =
ISFLG + 2
14
ISFLG =
ISFLG + 1
160
Q
P33



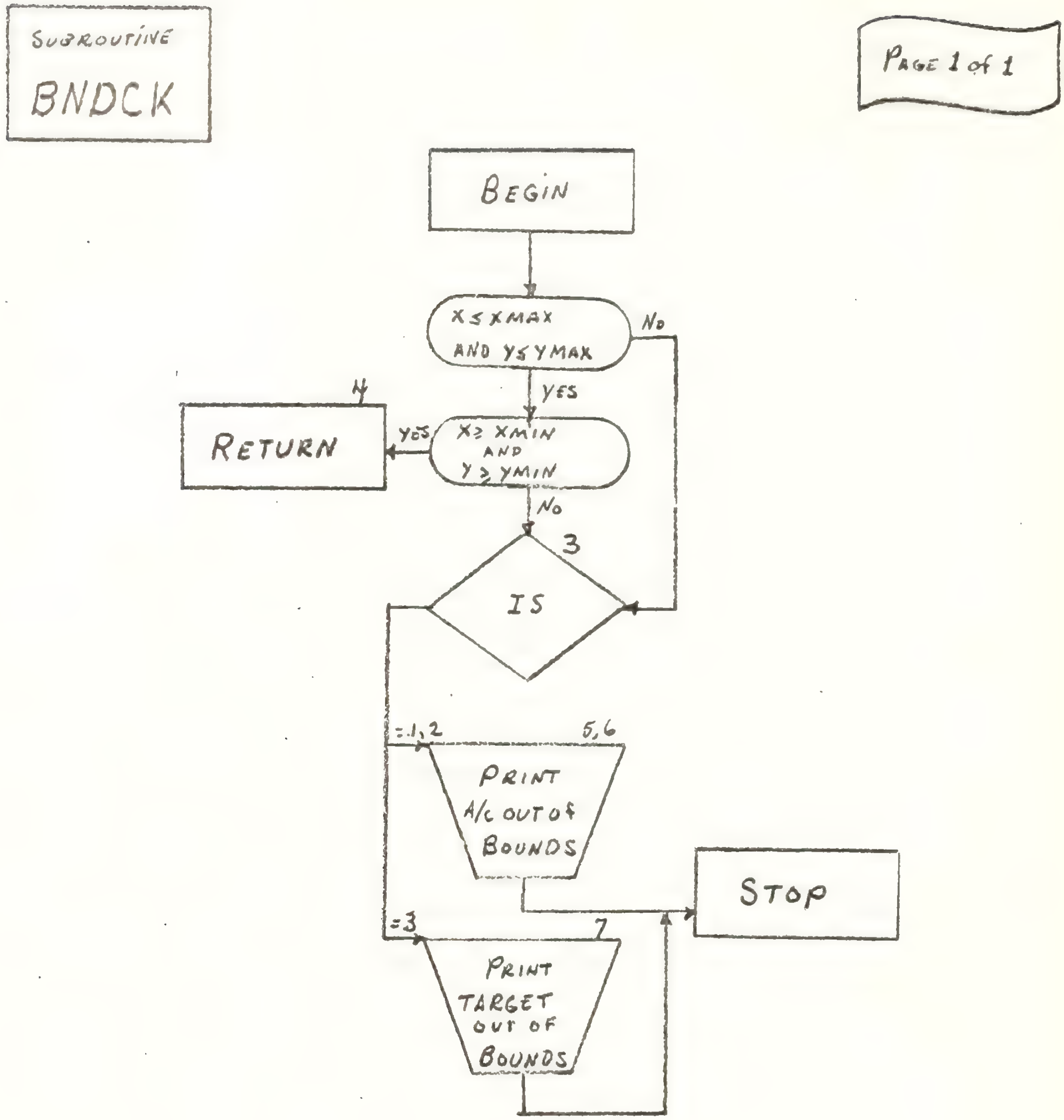
ARMREC
PAGE 10 OF 13
Ø
164
RANF(-1.) ≤ SPG(P3)
N
Y
166
IGFLG = 0
165
IGFLG = 2
167
RANF(-1.) ≤ SPT(P3)
N
Y
168
IGFLG = IGFLG + 1
169
IGFLG = IGFLG + 2
173
L
P37

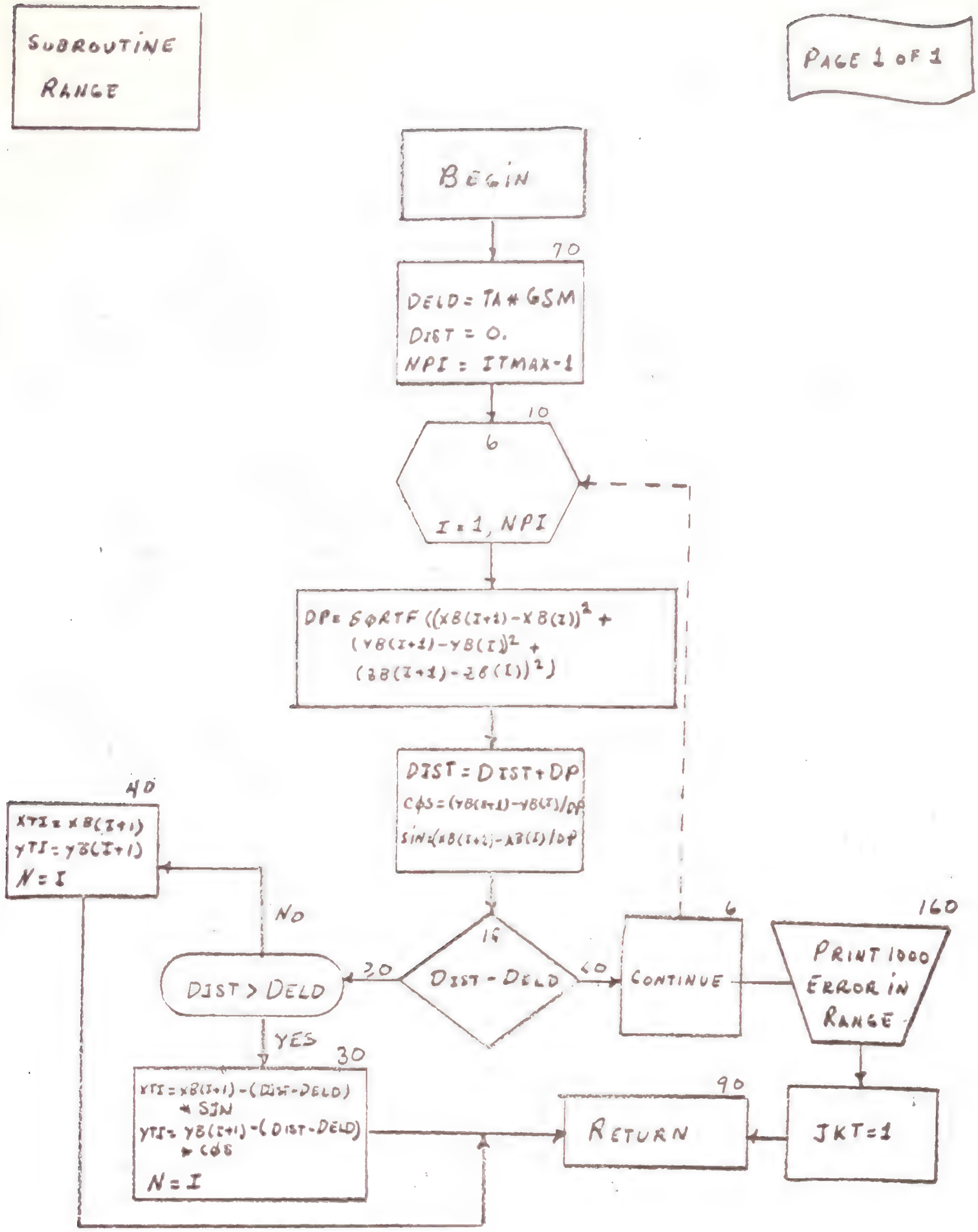

ARMREC
PAGE 11 OF 13
P
Q
19
T
P40
>0
IF
TTA-TI
-IPOP
300
=1,2
ISFLG
160
≤0
=3,4
21
201
Y
ISFLG=2
AND
IGFLG=2
N
TSV1 = TT
IPP = 0
R
PRINT
SIMULATION
STATUS
18
G
P33
CALL
ATTACK
691
CONTINUE
A
P29
22
CONTINUE
=2
JKT
=1
69
CALL
PRINT1
S
W

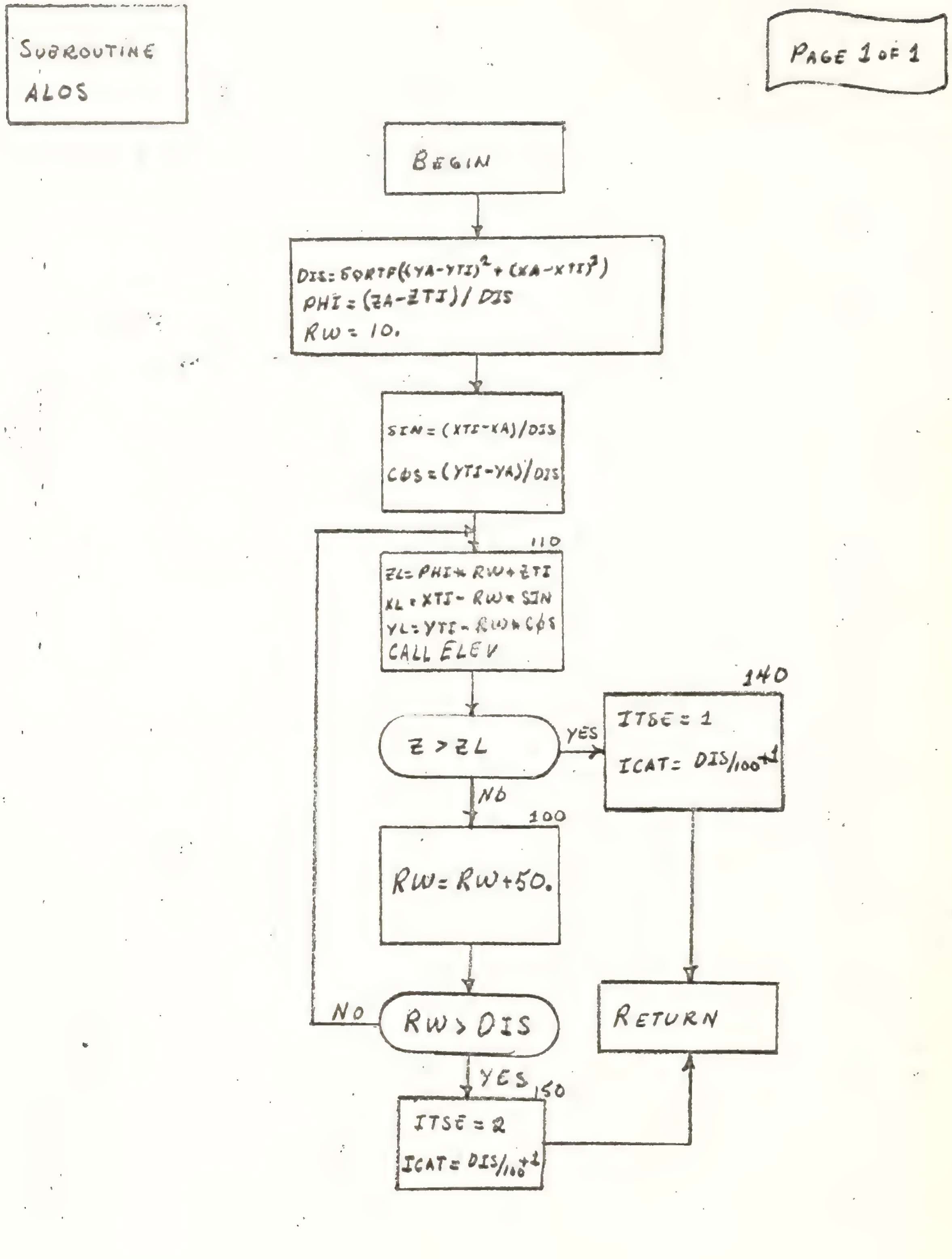

ARMREC
PAGE 12 OF 13
T
IF IPP
19
>0
≤0
IF IPP-I9
191
>0
≤0
190
PRINT 4040 A/c HAS POPPED UP
IPP=IPP+1 COMPUTE ZAI ZAII=ZAI
192
PRINT 4040
355
PRINT 4040
IPP=IPP+1 COMPUTE ZAI ZAII=ZAI
COMPUTE ZAI IPP=0
IF P2-1
≠0
23 U P41
20
CALL ALOS P2 = ITSE P1 = ICAT
121
T1 = TTA
IF P2-2
>0
<0
277
IGFLG=2
N
Y
21 R P39
22 W P39
IF RANF(-1.) - SPA(P1)
≤0
>0
194
ISFLG=2
195
ISFLG=0
196 V P41



ARMREC
PAGE 13 of 13
U
23
RANF(-1.)≤SPA(P1)
NO
27
ISFLG=2
AND
IGFLG=2
YES
22
W
P39
NO
21
R
P39
YES
240
ISFLG=ISFLG+2
TI=TTA
IPP=0
V
196
IF
RANF(-1.)-SPT(P1)
≤0
197
ISFLG=ISFLG+1
IPP=0
>0
198
ISFLG=ISFLG+2
IPP=0
199
ISFLG
=2
22
W
P39
=1,3,4
21
R
P39

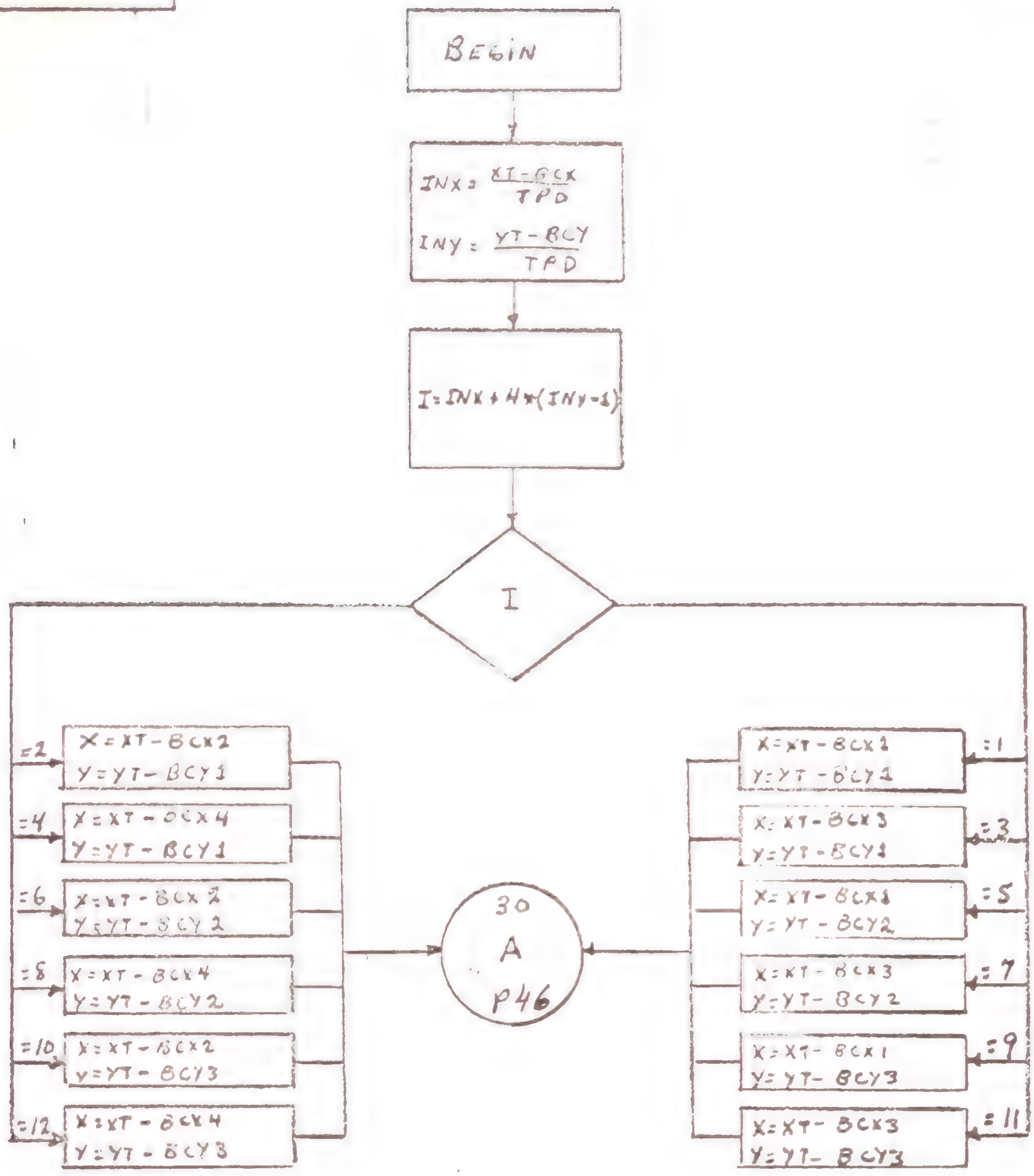

SUBROUTINE
BNDCK
PAGE 1 of 1
BEGIN
X ≤ XMAX AND Y ≤ YMAX
NO
YES
X ≥ XMIN AND Y ≥ YMIN
YES
4
RETURN
NO
3
IS
=1,2
5,6
PRINT A/C OUT OF BOUNDS
STOP
=3
7
PRINT TARGET OUT OF BOUNDS

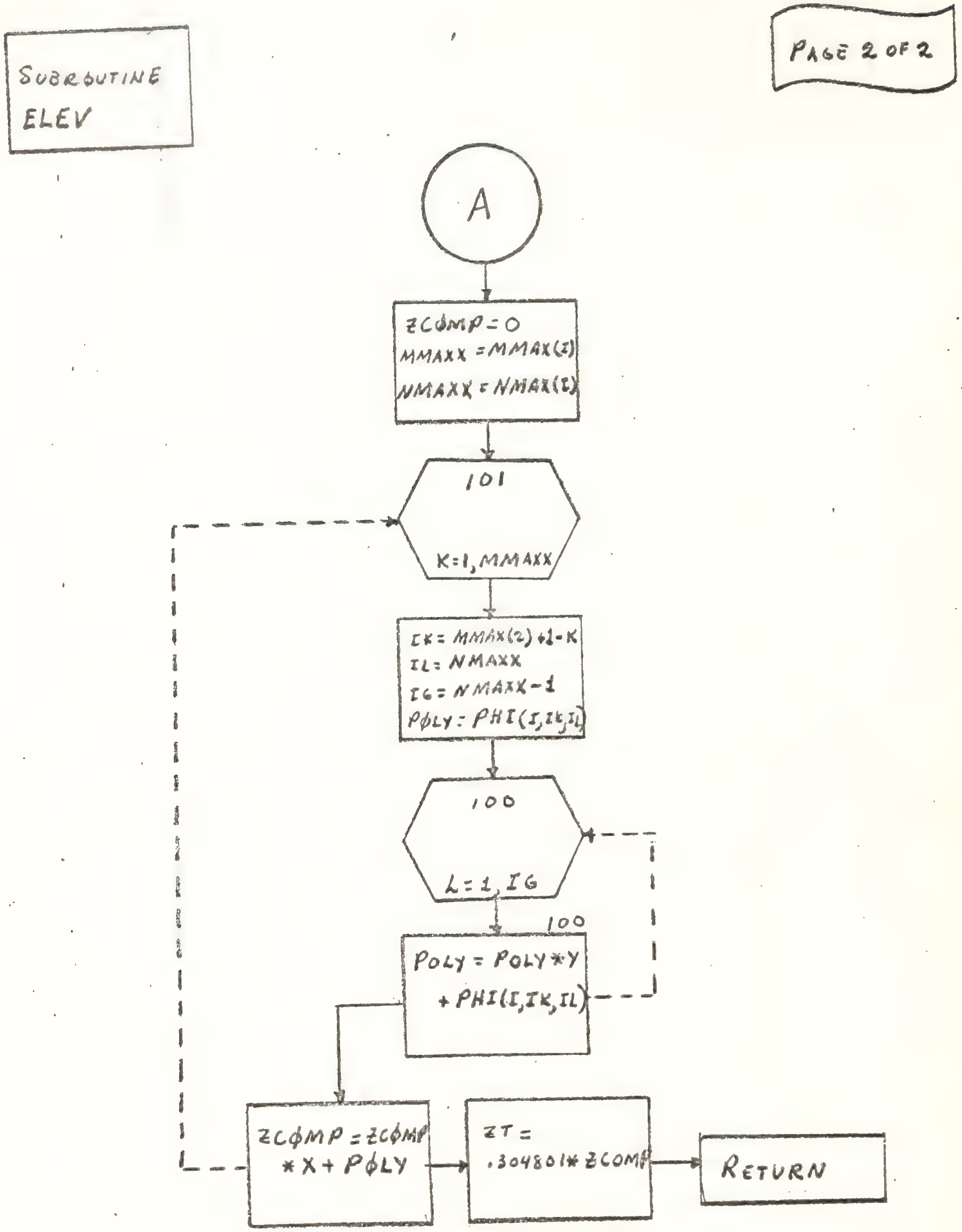

SUBROUTINE RANGE
PAGE 1 OF 1
BEGIN
70
DELD = TA * GSM
DIST = 0.
NPI = ITMAX-1
10
6
I = 1, NPI
DP = SQRTF ((XB(I+1) - XB(I))² + (YB(I+1) - YB(I))² + (ZB(I+1) - ZB(I))²)
DIST = DIST + DP
CØS = (YB(I+1) - YB(I))/DP
SIN = (XB(I+1) - XB(I))/DP
15
DIST - DELD
20
60
DIST > DELD
NO
YES
40
XTI = XB(I+1)
YTI = YB(I+1)
N = I
30
XTI = XB(I+1) - (DIST-DELD) * SIN
YTI = YB(I+1) - (DIST-DELD) * CØS
N = I
6
CONTINUE
160
PRINT 1000 ERROR IN RANGE
90
RETURN
JKT=1



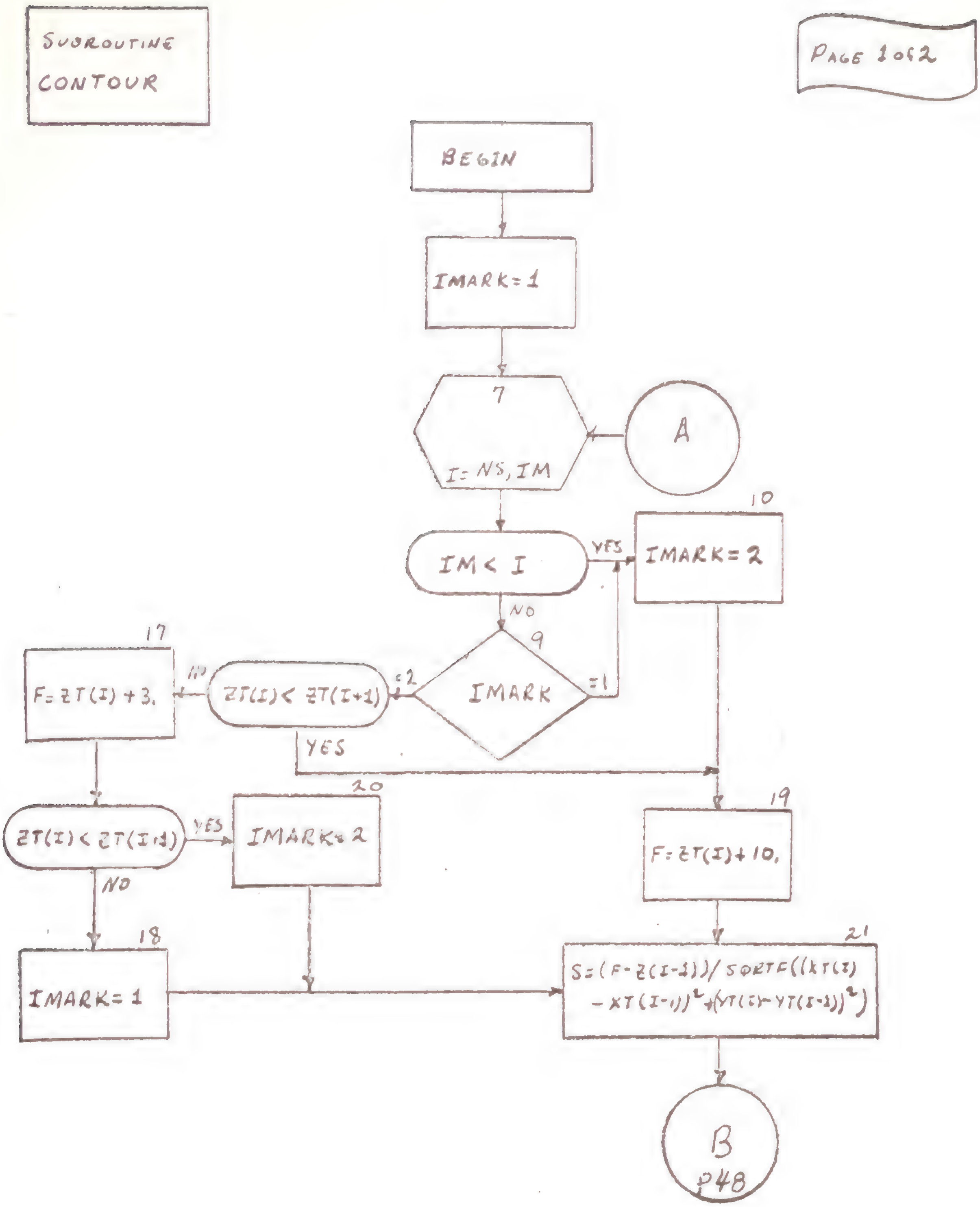
SUBROUTINE ALOS
PAGE 1 OF 1
BEGIN
DIS= SQRTF((YA-YTI)^2+(XA-XTI)^2)
PHI=(ZA-ZTI)/DIS
RW=10.
SIN=(XTI-XA)/DIS
COS=(YTI-YA)/DIS
110
ZL=PHI*RW+ZTI
XL=XTI-RW*SIN
YL=YTI-RW*COS
CALL ELEV
Z > ZL
YES
NO
140
ITSE=1
ICAT=DIS/100+1
100
RW=RW+50.
RW > DIS
NO
YES
150
ITSE=2
ICAT=DIS/100+1
RETURN

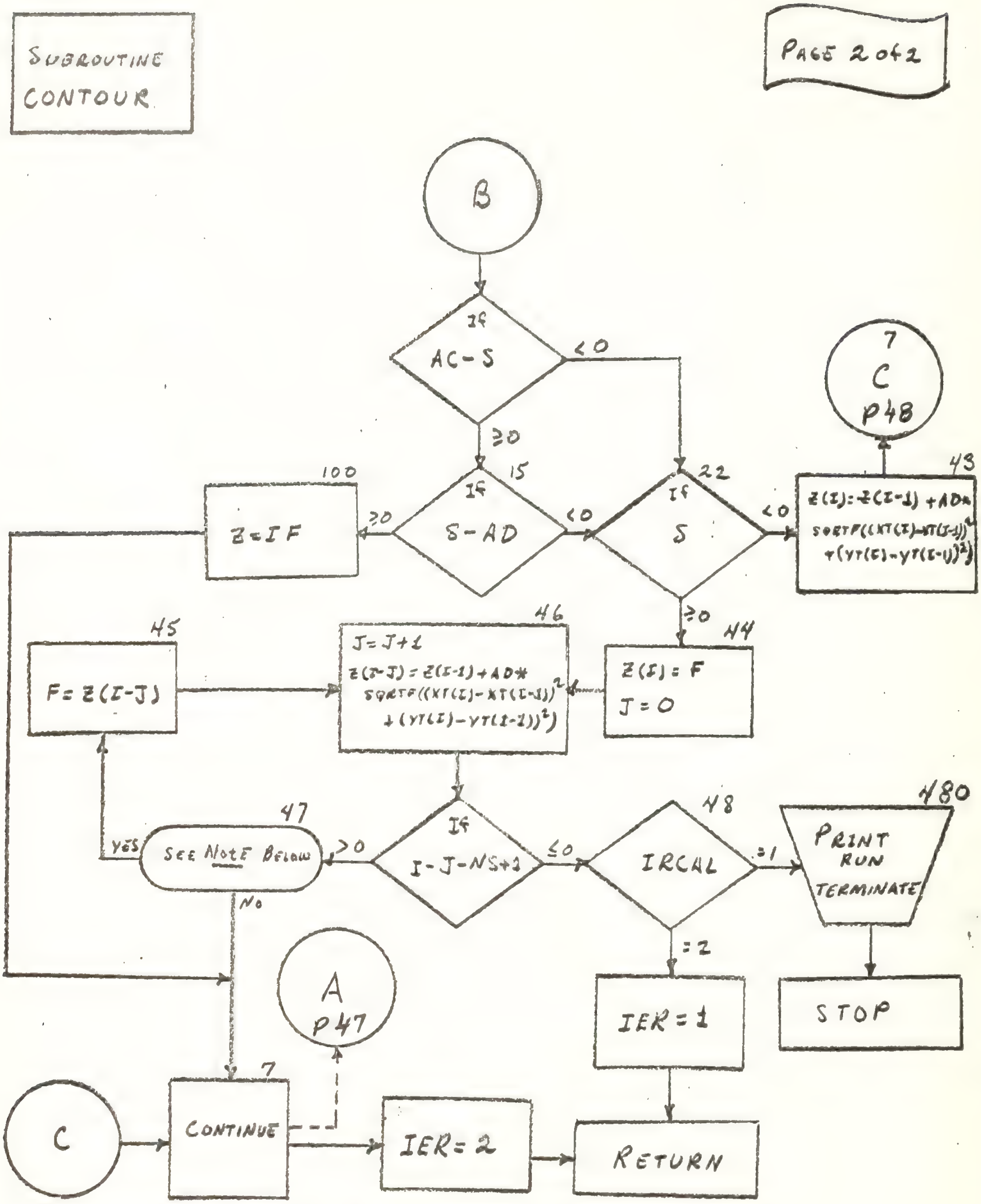
SUBROUTINE ELEV





SUBROUTINE ELEV
PAGE 2 OF 2
A
ZCØMP = 0
MMAXX = MMAX(I)
NMAXX = NMAX(I)
101
K=1, MMAXX
IK = MMAX(I)+1-K
IL = NMAXX
IG = NMAXX-1
PØLY = PHI(I,IK,IL)
100
L=1, IG
100
POLY = POLY*Y + PHI(I,IK,IL)
ZCØMP = ZCØMP *X + PØLY
ZT = .304801*ZCOMP
RETURN

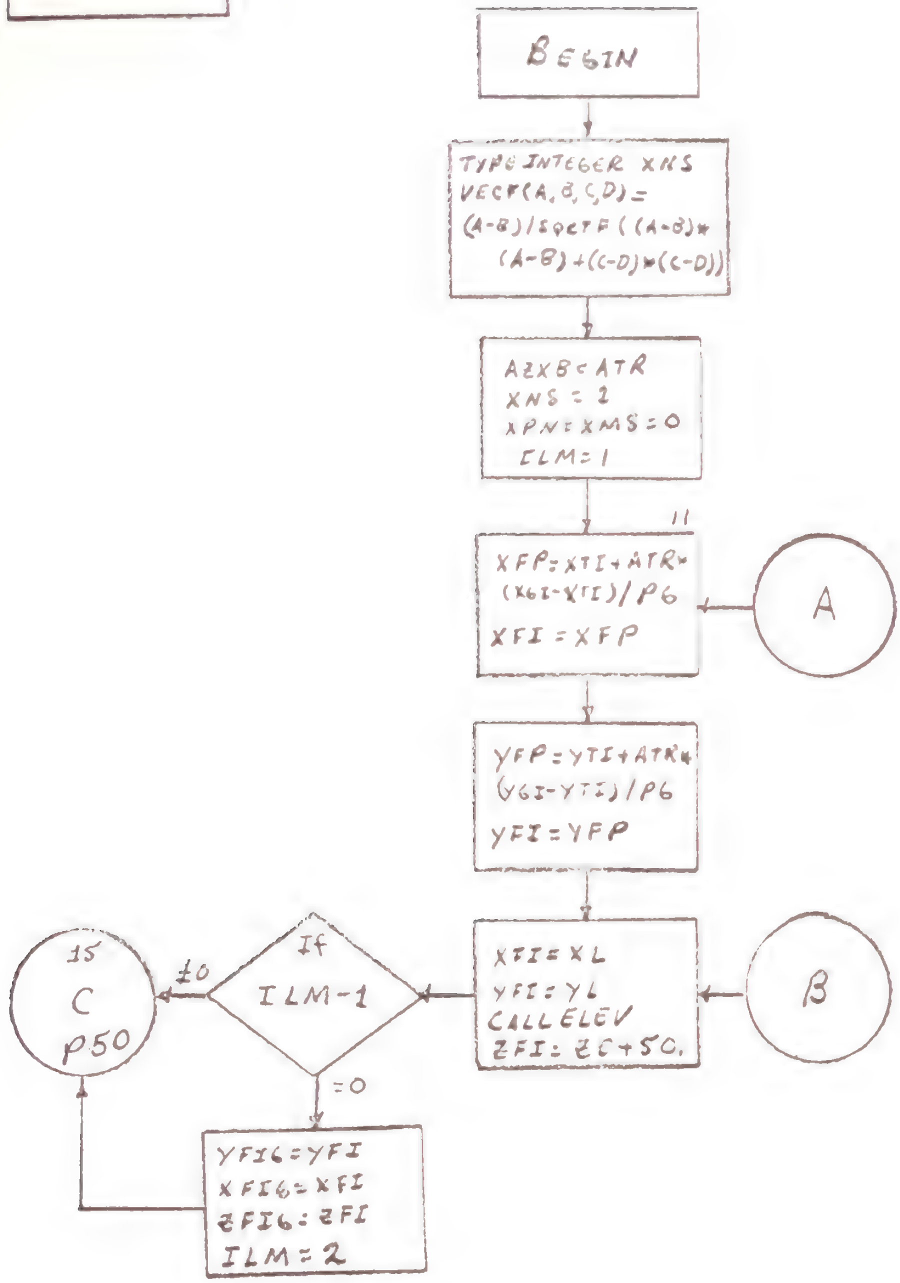

SUBROUTINE CONTOUR
PAGE 1 OF 2
BEGIN
IMARK=1
7
I=NS,IM
A
IM< I
YES
NO
10
IMARK=2
9
IMARK
=1
=2
ZT(I)< ZT(I+1)
NO
YES
17
F=ZT(I)+3.
ZT(I)< ZT(I-1)
YES
NO
20
IMARK=2
18
IMARK=1
19
F=ZT(I)+10.
21
S=(F-Z(I-1))/SQRTF((XT(I)-XT(I-1))^2+(YT(I)-YT(I-1))^2)
B
P48

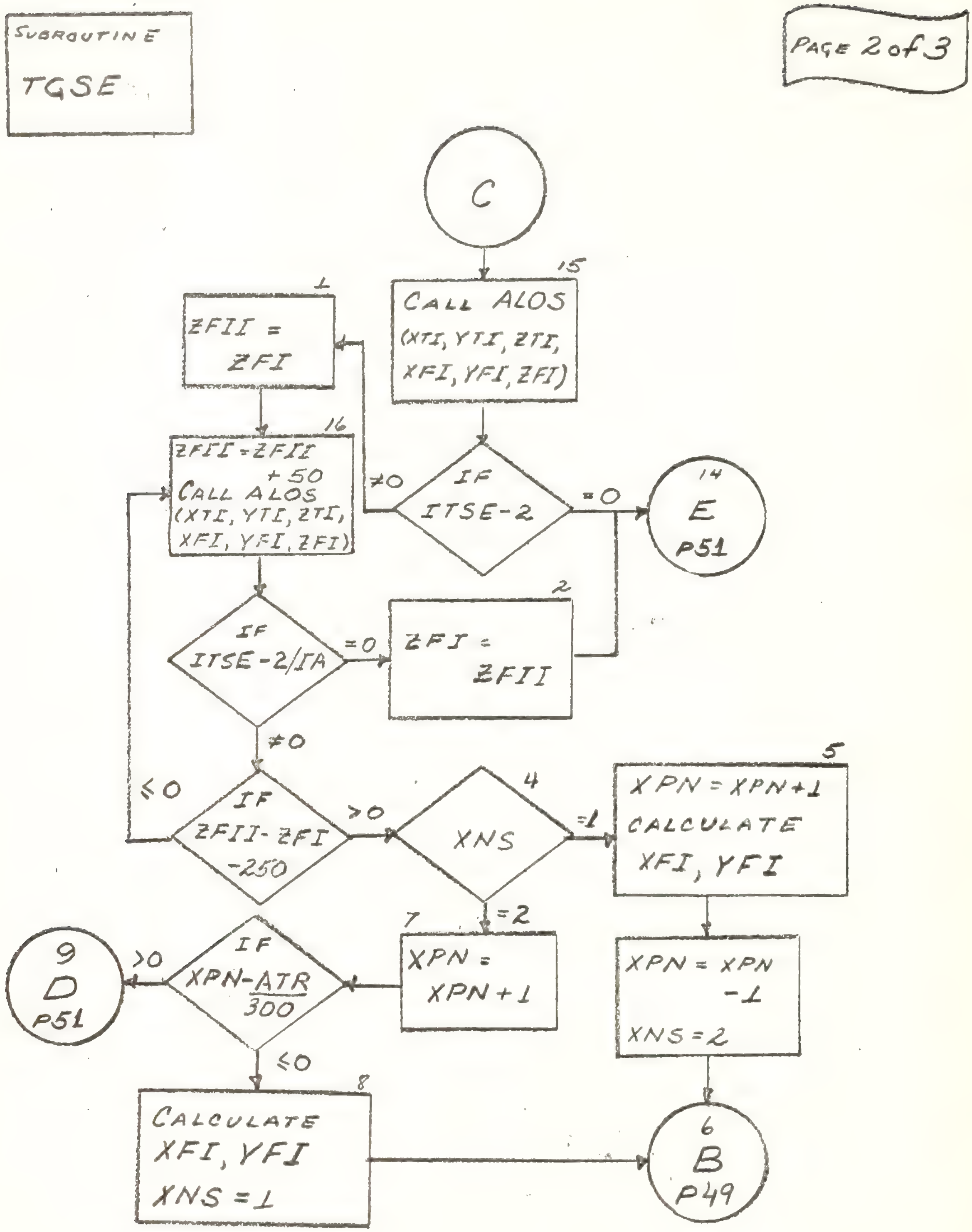


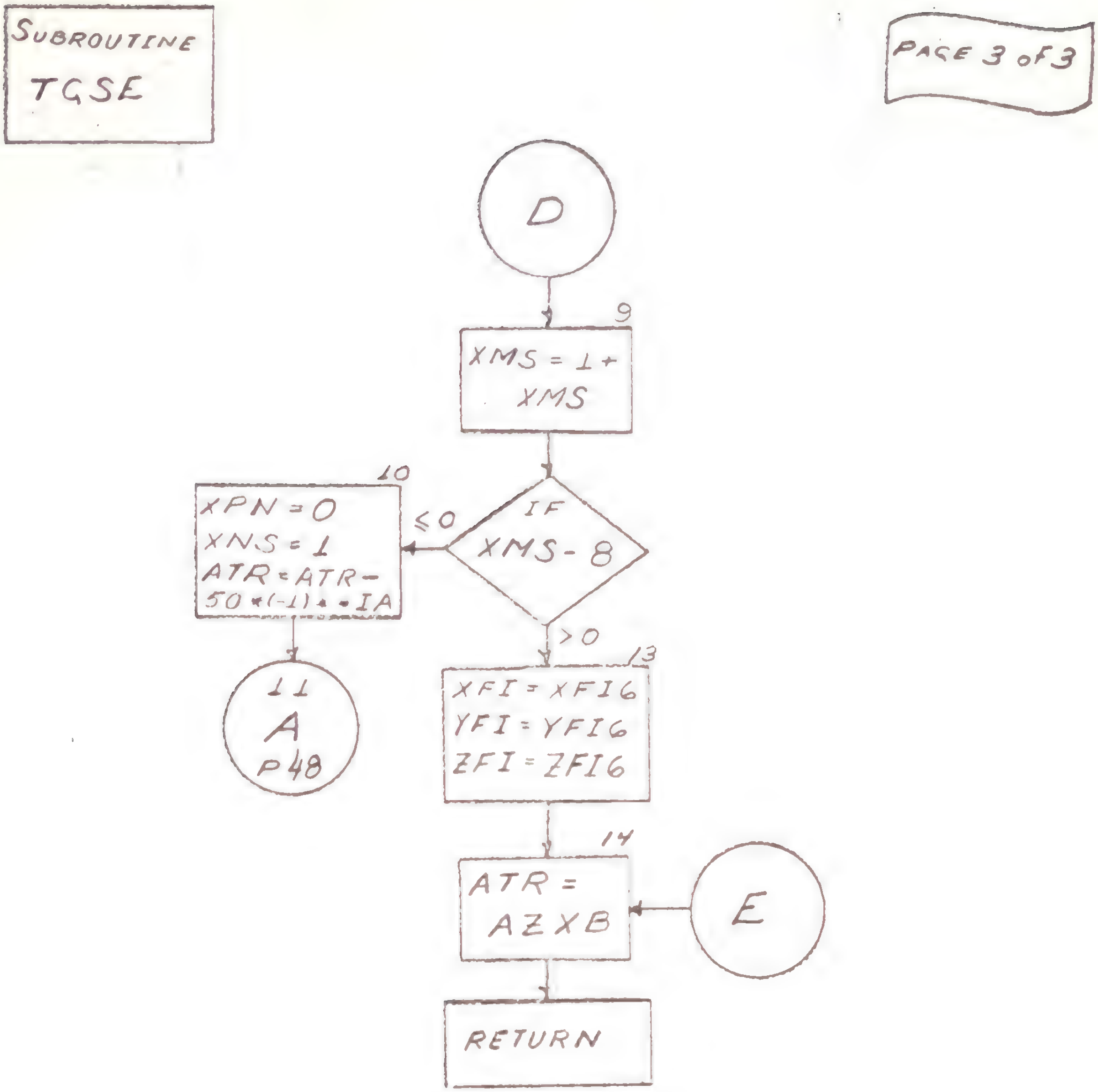
NOTE ; EXPRESSION FOR STATEMENT #47

(((Z(I-J) - Z(I-J-1) / SQRTF((XT(I-J) - XT(I-J-1))^2 + (YT(I-J) - YT(I-J-1))^2))

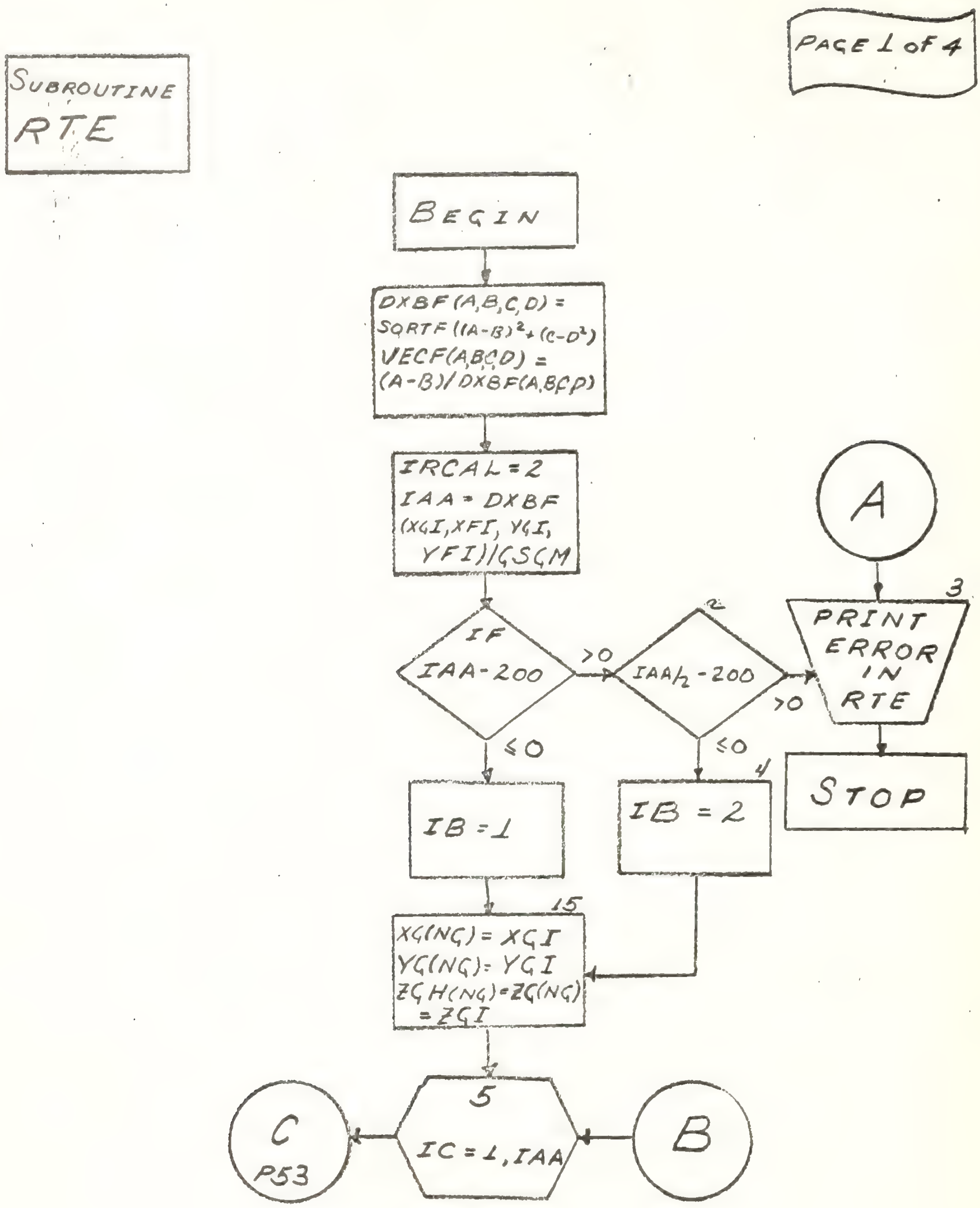
SUBROUTINE TGSE

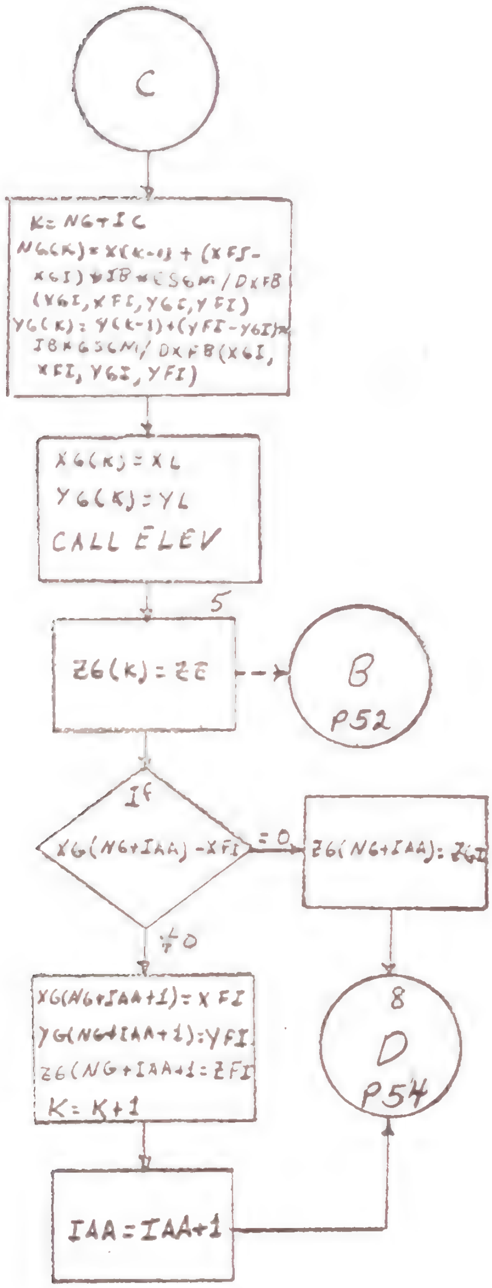


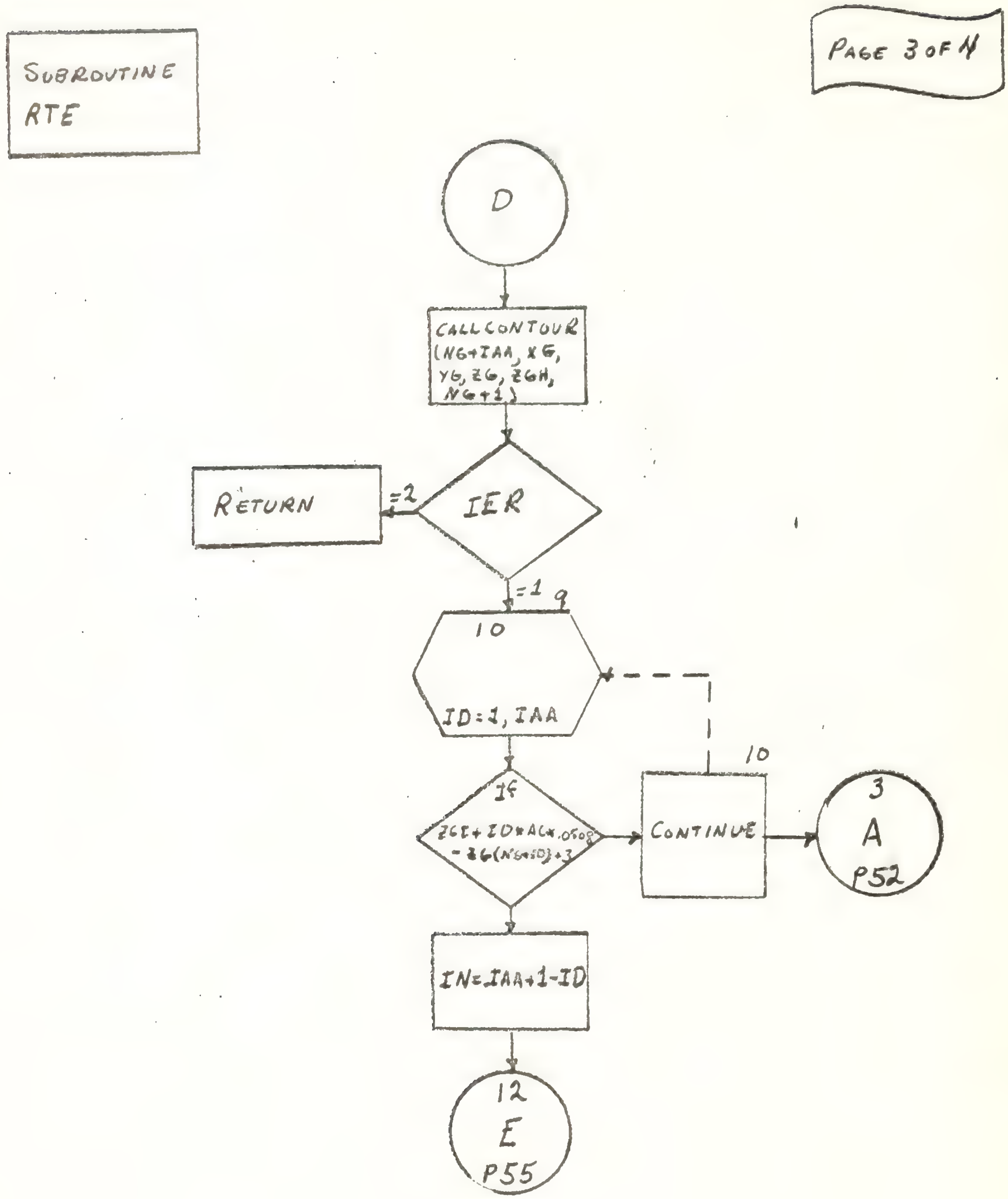

SUBROUTINE
TGSE
PAGE 2 of 3
C
15
CALL ALOS
(XTI, YTI, ZTI,
XFI, YFI, ZFI)
1
ZFII =
ZFI
16
ZFII = ZFII
+ 50
CALL ALOS
(XTI, YTI, ZTI,
XFI, YFI, ZFI)
≠0
IF
ITSE-2
=0
14
E
P51
IF
ITSE-2/IA
=0
2
ZFI =
ZFII
≠0
≤0
IF
ZFII-ZFI
-250
>0
4
XNS
=1
5
XPN = XPN+1
CALCULATE
XFI, YFI
=2
7
XPN =
XPN + 1
9
D
P51
>0
IF
XPN-ATR/300
≤0
XPN = XPN
-1
XNS = 2
8
CALCULATE
XFI, YFI
XNS = 1
6
B
P49



SUBROUTINE
TGSE
PAGE 3 of 3
D
9
XMS = 1 +
XMS
10
XPN = 0
XNS = 1
ATR = ATR -
50 * (-1) * * IA
≤ 0
IF
XMS - 8
> 0
11
A
P48
13
XFI = XFI6
YFI = YFI6
ZFI = ZFI6
14
ATR =
AZXB
E
RETURN

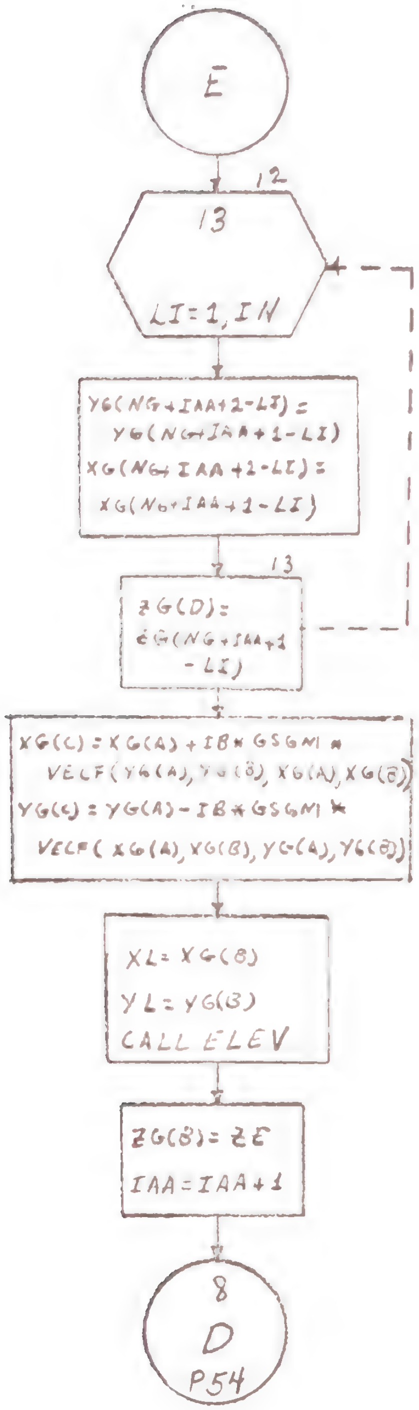

SUBROUTINE RTE
PAGE 1 of 4
BEGIN
DXBF(A,B,C,D) = SQRTF((A-B)²+(C-D²))
VECF(A,B,C,D) = (A-B)/DXBF(A,B,C,D)
IRCAL=2
IAA = DXBF (XGI, XFI, YGI, YFI)/GSGM
IF IAA-200
>0
≤0
2
IAA/2 - 200
>0
≤0
A
3
PRINT ERROR IN RTE
STOP
IB = 1
4
IB = 2
15
XG(NG) = XGI
YG(NG) = YGI
ZGH(NG)=ZG(NG) = ZGI
5
IC = 1, IAA
C
P53
B


C
K= NG+IC
NG(K)= X(K-1) + (XFI-XGI)*IB*GSGM/DXFB(XGI,XFI,YGI,YFI)
YG(K)= Y(K-1)+(YFI-YGI)*IB*GSGM/DXFB(XGI,XFI,YGI,YFI)
XG(K)=XL
YG(K)=YL
CALL ELEV
5
ZG(K)=ZE
B
P52
IF
XG(NG+IAA)-XFI
=0
≠0
ZG(NG+IAA)=ZGI
XG(NG+IAA+1)=XFI
YG(NG+IAA+1)=YFI
ZG(NG+IAA+1=ZFI
K= K+1
IAA=IAA+1
8
D
P54

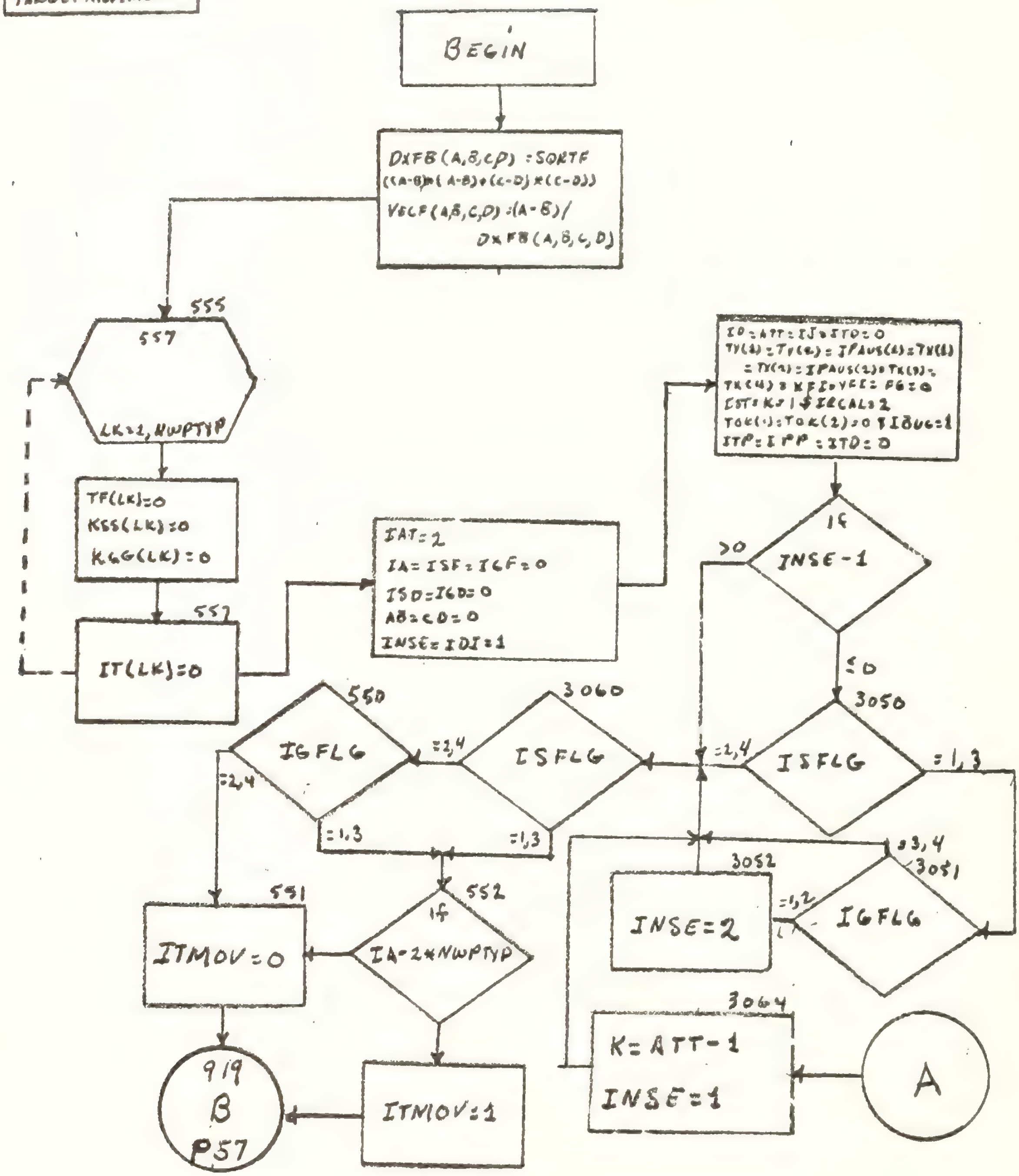


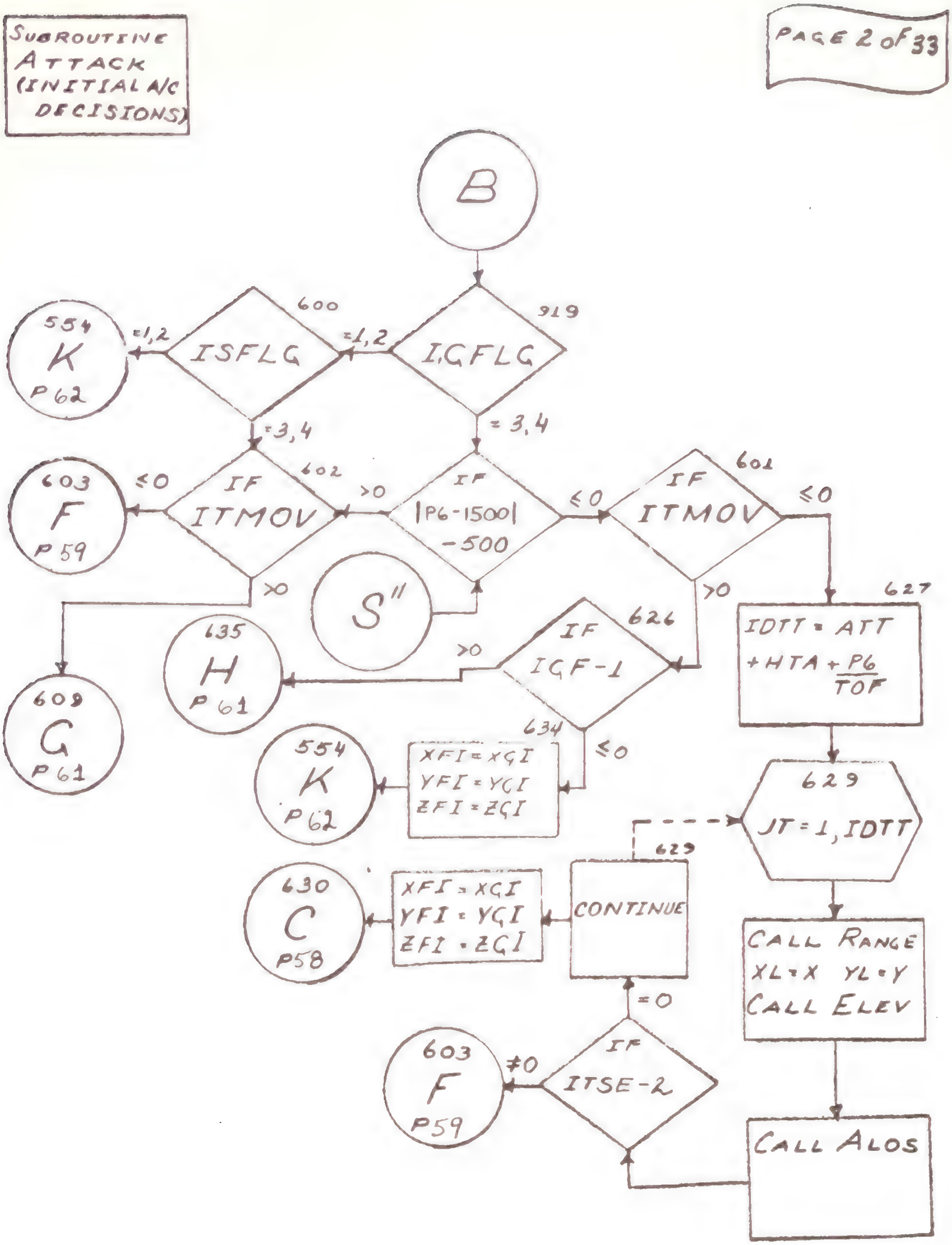
SUBROUTINE RTE
PAGE 3 OF 4
D
CALL CONTOUR (NG+IAA, XG, YG, ZG, ZGH, NG+1)
IER
=2
RETURN
=1
9
10
ID=1, IAA
IF
ZGI+ID*AC*.0508 -ZG(NG+ID)+3
10
CONTINUE
3
A
P52
IN=IAA+1-ID
12
E
P55

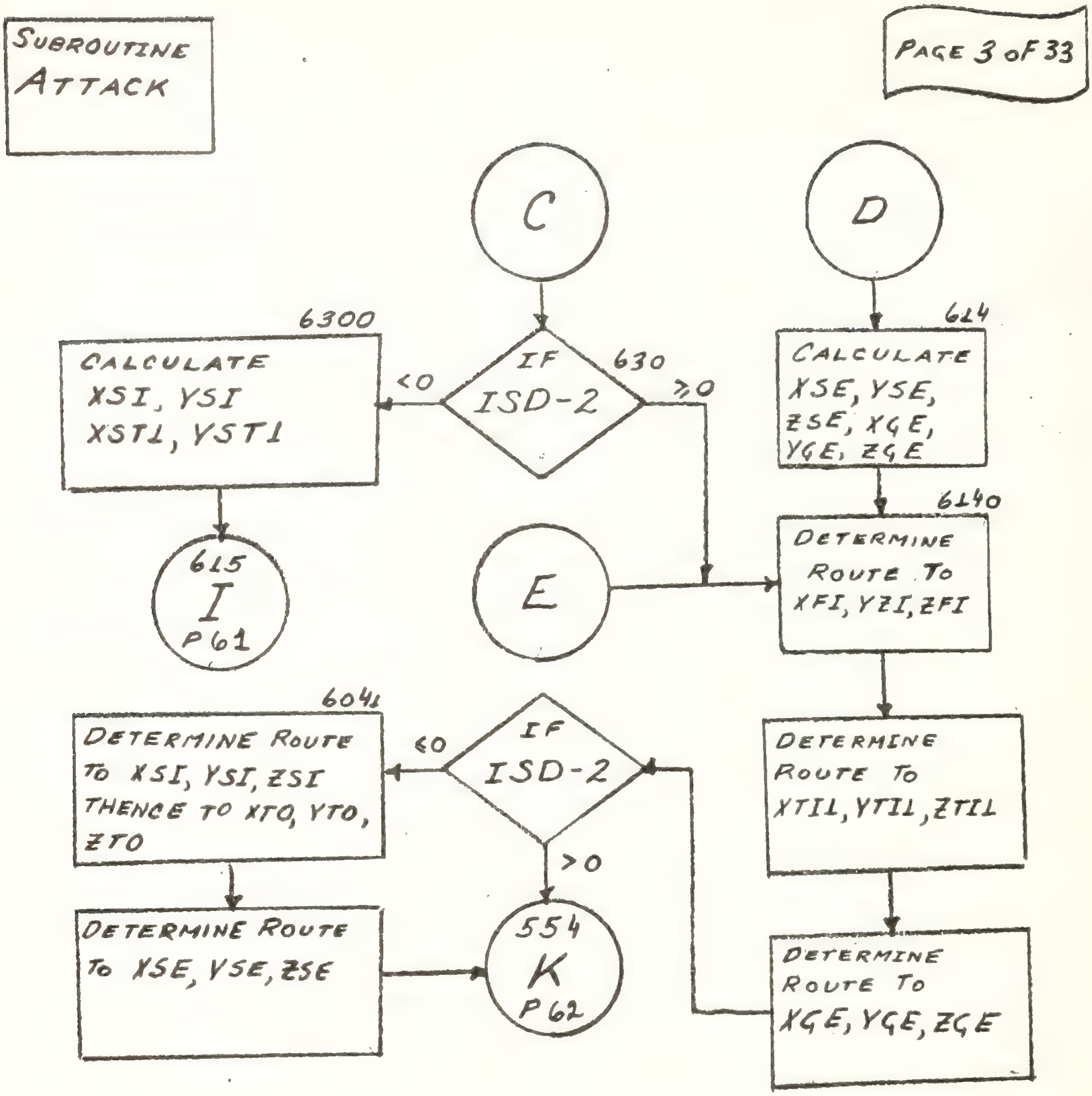
SUBROUTINE RTE

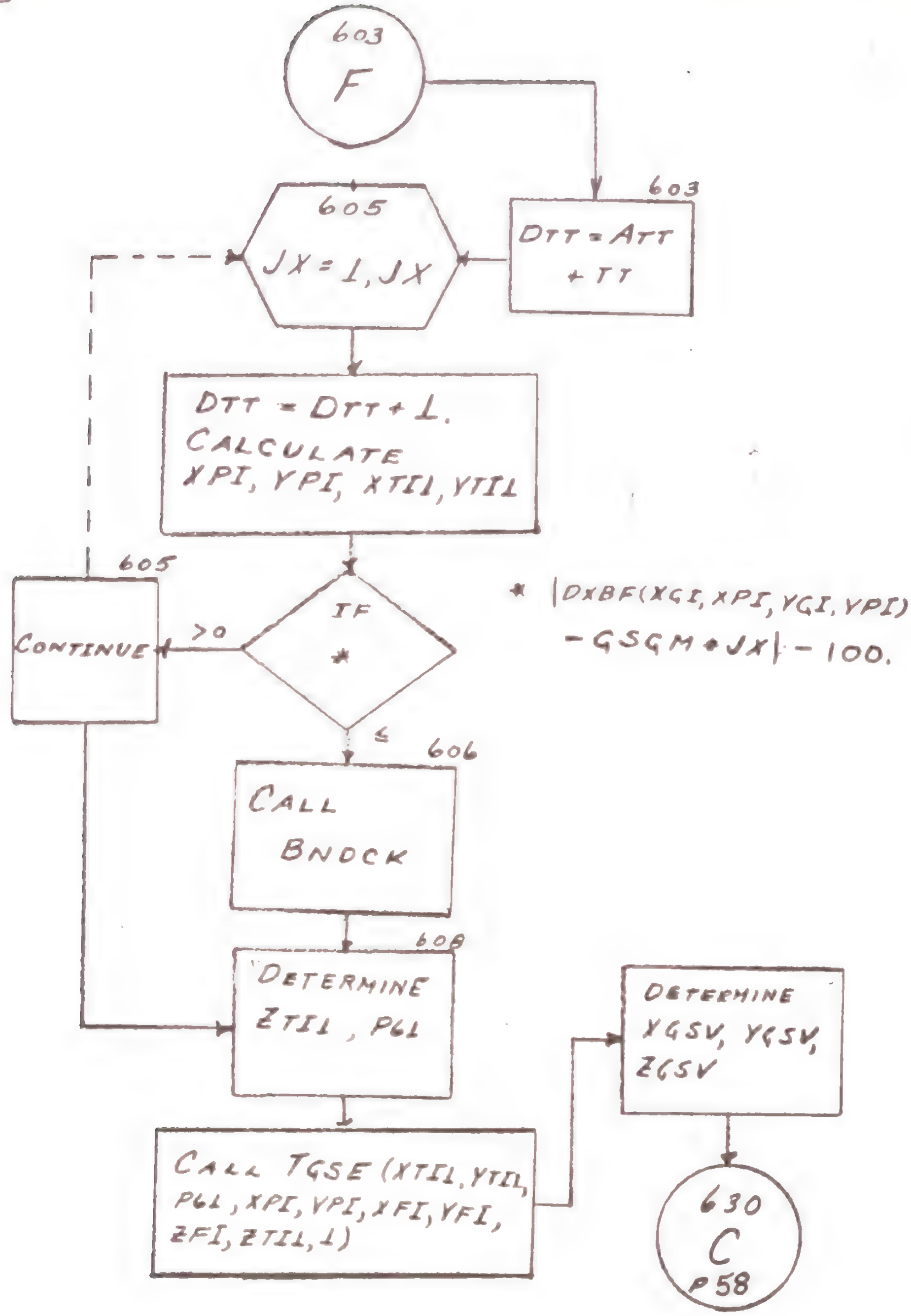


SUBROUTINE
ATTACK
INITIALIZATION AND
TARGET MOVEMENT



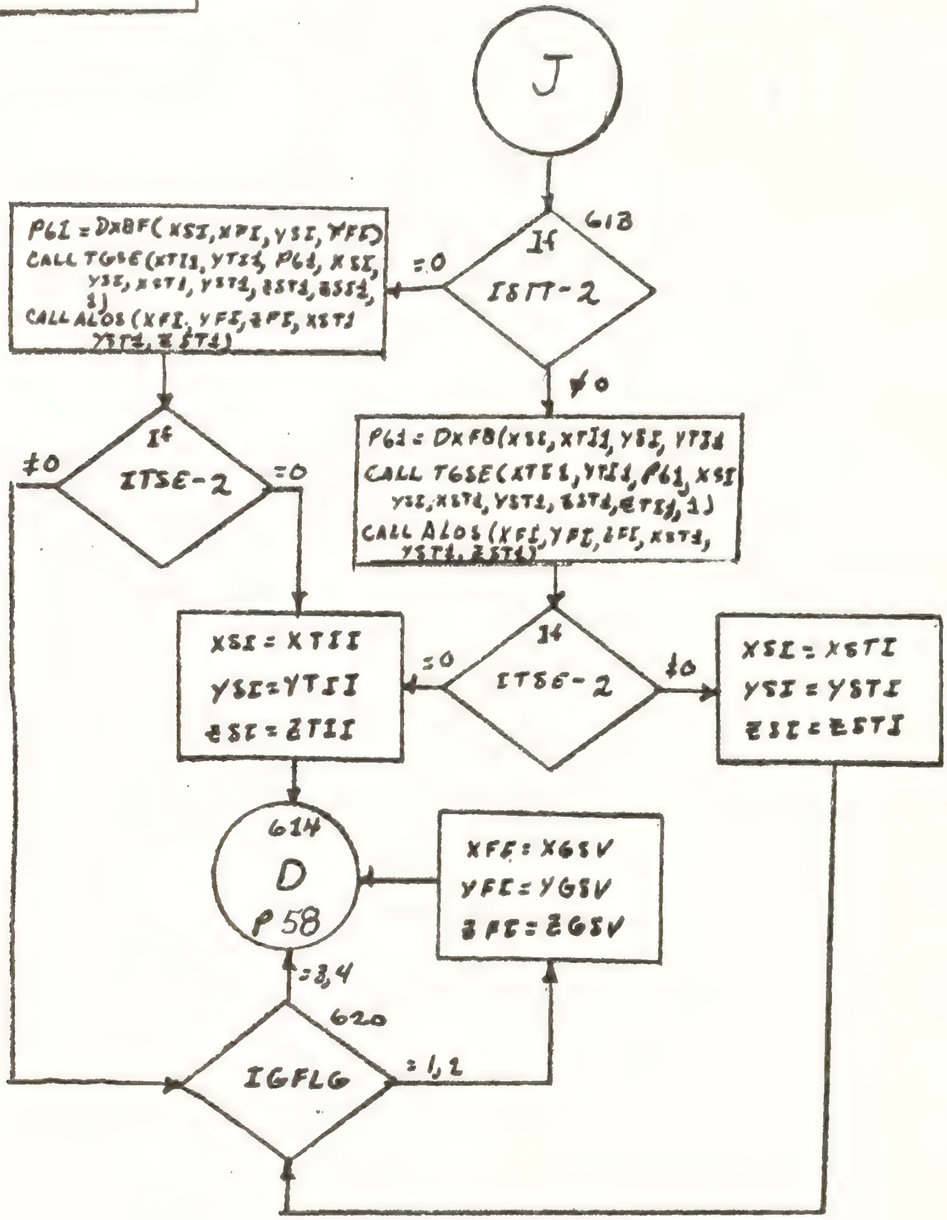

SUBROUTINE ATTACK (INITIAL A/C DECISIONS)
PAGE 2 of 33
B
919
IGFLG
=1,2
=3,4
600
ISFLG
554
K
P62
602
IF ITMOV
≤0
>0
603
F
P59
609
G
P61
IF |P6-1500| -500
S''
601
IF ITMOV
627
IDTT = ATT + HTA + P6/TOF
626
IF IGF-1
135
H
P61
634
XFI = XGI
YFI = YGI
ZFI = ZGI
629
JT = 1, IDTT
CALL RANGE
XL = X YL = Y
CALL ELEV
CALL ALOS
CONTINUE
630
C
P58
IF ITSE-2
≠0
=0

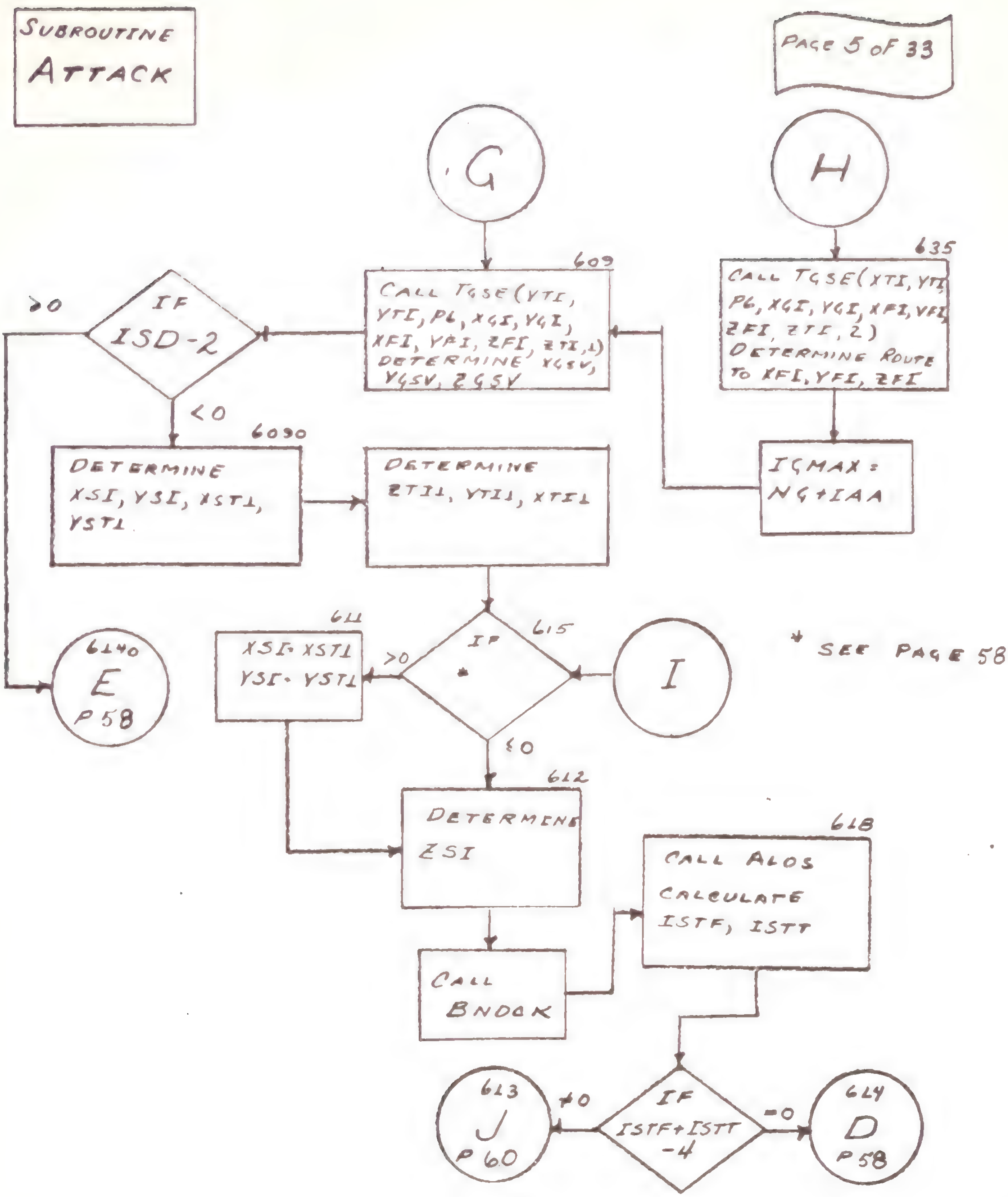

SUBROUTINE ATTACK
PAGE 3 OF 33
C
D
IF ISD-2
630
<0
≥0
6300
CALCULATE XSI, YSI XST1, YST1
614
CALCULATE XSE, YSE, ZSE, XGE, YGE, ZGE
6140
DETERMINE ROUTE TO XFI, YZI, ZFI
615
I
P61
E
6041
DETERMINE ROUTE TO XSI, YSI, ZSI THENCE TO XTO, YTO, ZTO
IF ISD-2
<0
>0
DETERMINE ROUTE TO XTI1, YTI1, ZTI1
DETERMINE ROUTE TO XSE, YSE, ZSE
554
K
P62
DETERMINE ROUTE TO XGE, YGE, ZGE

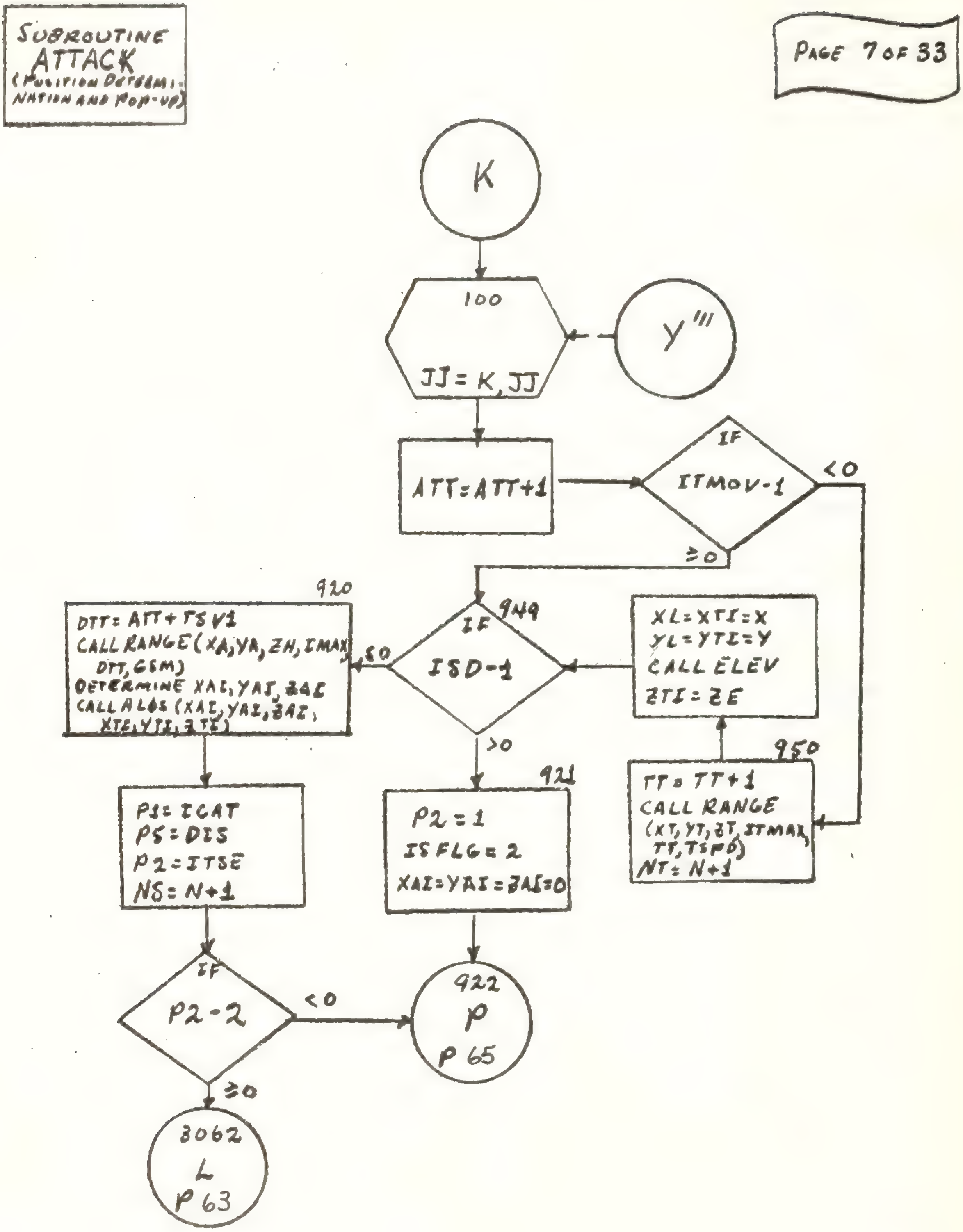

603
F
603
DTT = ATT + TT
605
JX = 1, JX
DTT = DTT + 1.
CALCULATE
XPI, YPI, XTI1, YTI1
605
CONTINUE
>0
IF
*
* |DXBF(XGI, XPI, YGI, YPI) - GSGM * JX| - 100.
≤
606
CALL
BNOCK
608
DETERMINE
ZTI1 , PG1
CALL TGSE (XTI1, YTI1, PG1, XPI, YPI, XFI, YFI, ZFI, ZTI1, 1)
DETERMINE
XGSV, YGSV, ZGSV
630
C
P 58

SUBROUTINE

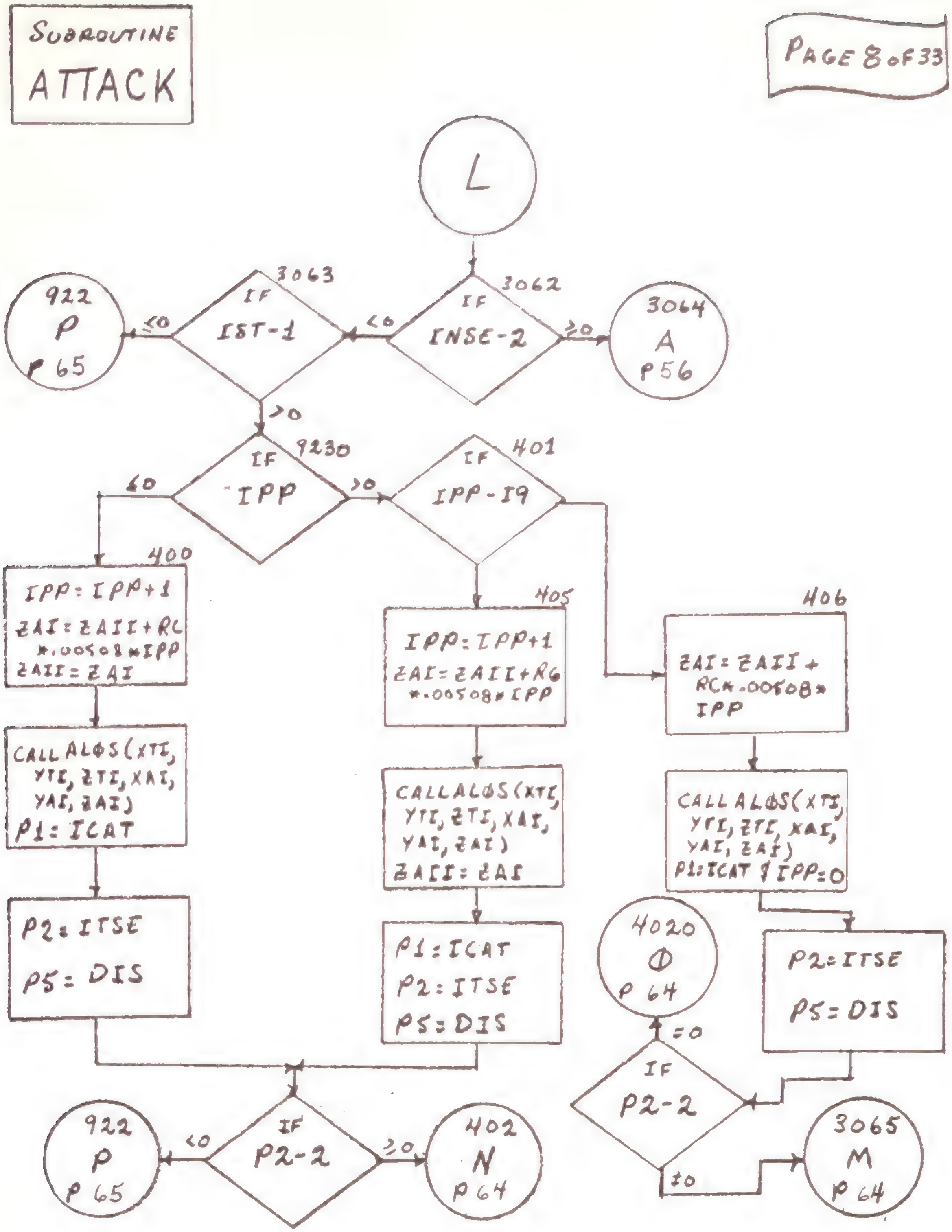
ATTACK



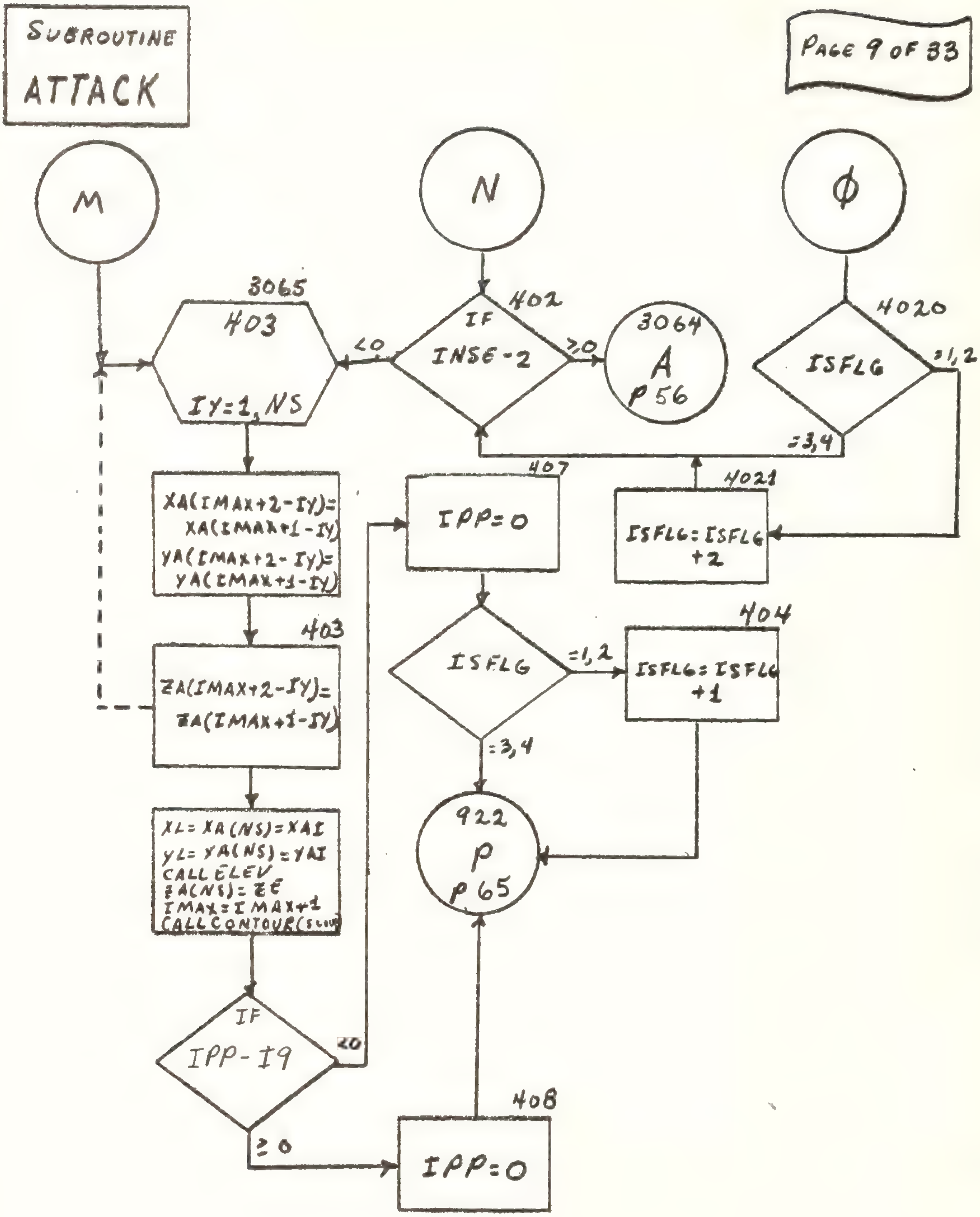

SUBROUTINE ATTACK
PAGE 5 OF 33
G
H
609
CALL TGSE(YTI, YTI, P6, XGI, YGI, XFI, YFI, ZFI, ZTI, 1) DETERMINE XGSV, YGSV, ZGSV
635
CALL TGSE(XTI, YTI, P6, XGI, YGI, XFI, YFI, ZFI, ZTI, 2) DETERMINE ROUTE TO XFI, YFI, ZFI
IF ISD-2
>0
<0
6090
DETERMINE XSI, YSI, XST1, YST1
DETERMINE ZTI1, YTI1, XTI1
IGMAX = NG+IAA
6140
E
P 58
611
XSI = XST1
YSI = YST1
>0
615
IF *
≤0
I
* SEE PAGE 58
612
DETERMINE ZSI
618
CALL ALOS
CALCULATE ISTF, ISTT
CALL BNOCK
613
J
P 60
≠0
IF ISTF+ISTT -4
=0
614
D
P 58

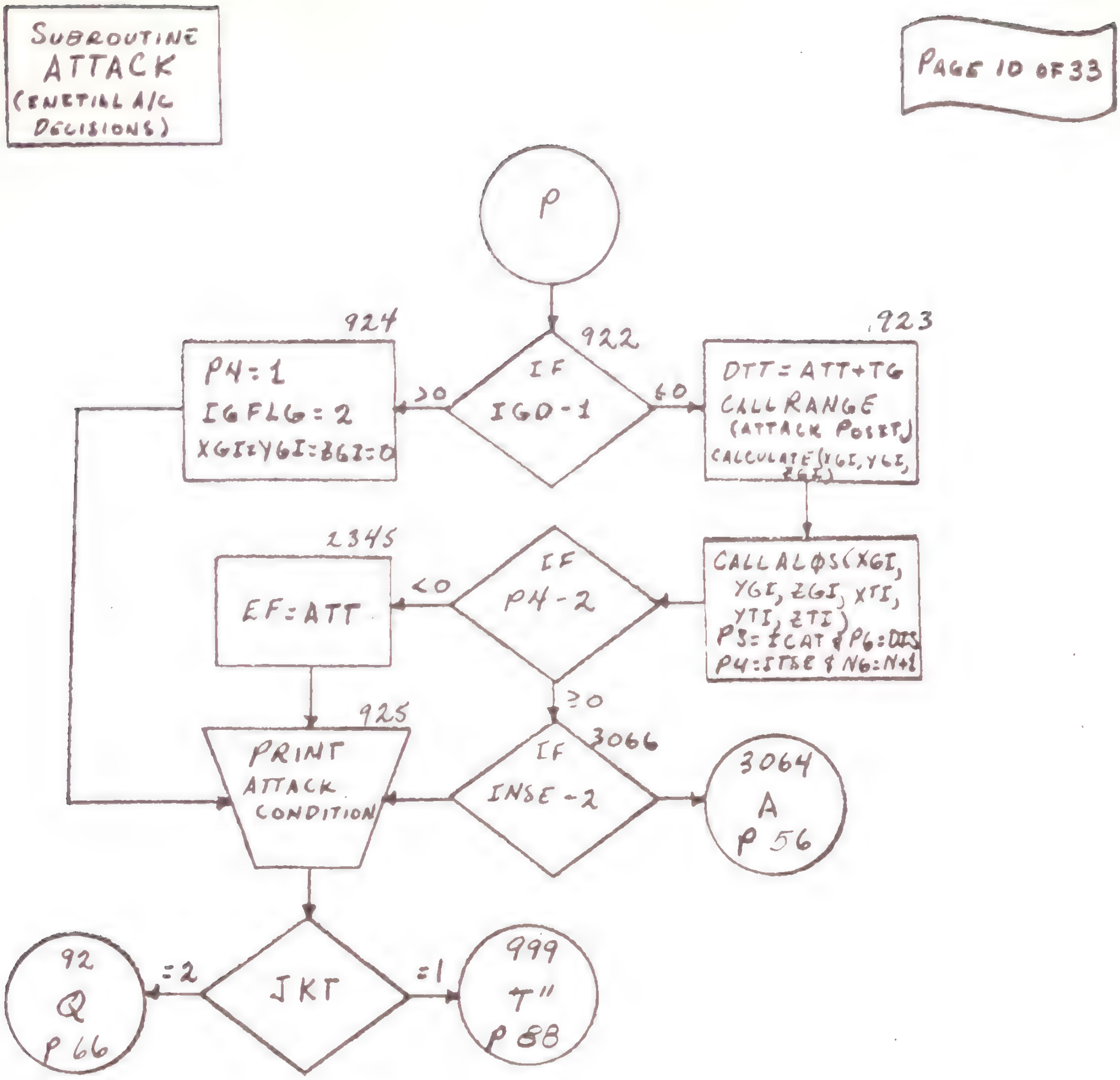


SUBROUTINE ATTACK (POSITION DETERMINATION AND POP-UP)
PAGE 7 OF 33
K
100
JJ = K, JJ
Y'''
ATT=ATT+1
IF
ITMOV-1
<0
≥0
920
DTT= ATT+TSV1
CALL RANGE(XA,YA,ZH,IMAX, DTT,GSM)
DETERMINE XAI,YAI,ZAI
CALL ALOS (XAI,YAI,ZAI, XTI,YTI,ZTI)
≤0
949
IF
ISD-1
>0
XL=XTI=X
YL=YTI=Y
CALL ELEV
ZTI=ZE
950
TT=TT+1
CALL RANGE (XT,YT,ZT,ITMAX, TT,TSPD)
NT=N+1
921
P2=1
ISFLG=2
XAI=YAI=ZAI=0
P1=ICAT
P5=DIS
P2=ITSE
NS=N+1
IF
P2-2
<0
≥0
922
P
P 65
3062
L
P 63

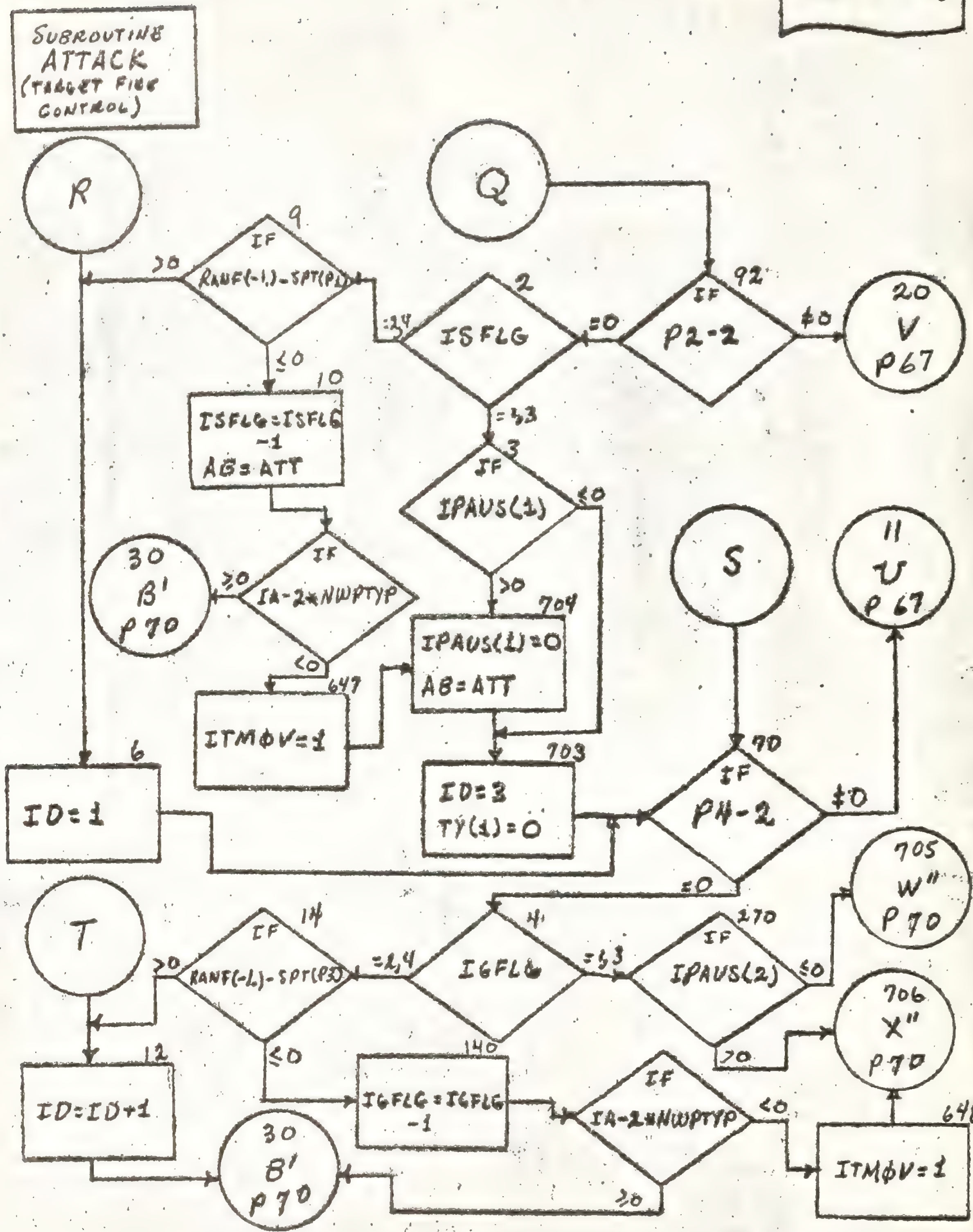


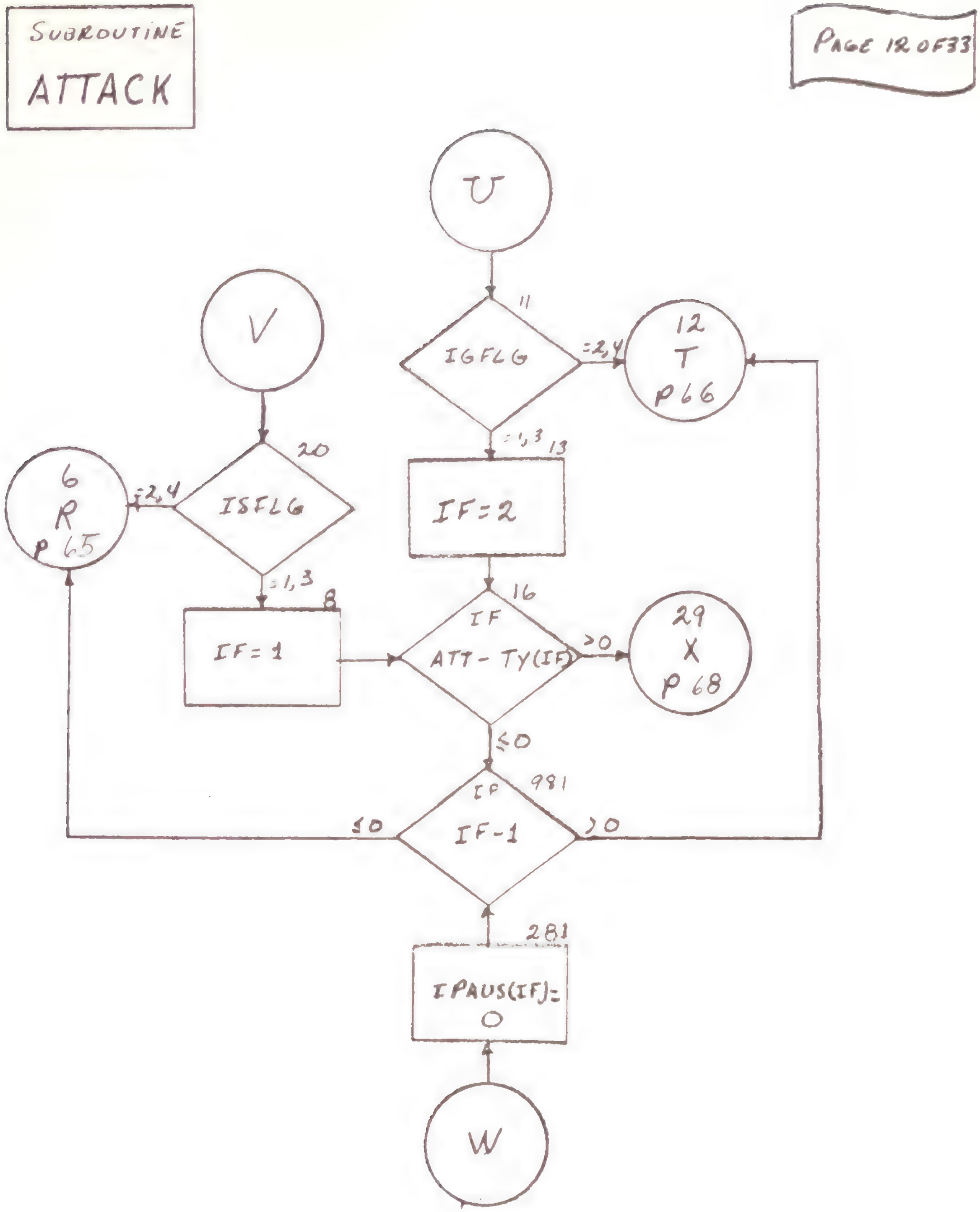
SUBROUTINE
ATTACK
PAGE 8 OF 33
L
3062
IF
INSE-2
≥0
3064
A
P56
<0
3063
IF
IST-1
≤0
922
P
P 65
>0
9230
IF
IPP
≤0
>0
401
IF
IPP-I9
400
IPP=IPP+1
ZAI=ZAII+RC
*.00508*IPP
ZAII=ZAI
CALL ALØS(XTI,
YTI, ZTI, XAI,
YAI, ZAI)
P1=ICAT
P2=ITSE
P5=DIS
405
IPP=IPP+1
ZAI=ZAII+RG
*.00508*IPP
CALLALØS(XTI,
YTI, ZTI, XAI,
YAI, ZAI)
ZAII=ZAI
P1=ICAT
P2=ITSE
P5=DIS
406
ZAI=ZAII+
RC*.00508*
IPP
CALL ALØS(XTI,
YTI, ZTI, XAI,
YAI, ZAI)
P1=ICAT $ IPP=0
P2=ITSE
P5=DIS
4020
Φ
P 64
=0
IF
P2-2
≠0
3065
M
P 64
922
P
P 65
<0
IF
P2-2
≥0
402
N
P64


SUBROUTINE ATTACK
PAGE 9 OF 33
M
N
Φ
3065
403
IY=1, NS
402
IF INSE-2
<0
≥0
3064
A
P 56
4020
ISFLG
=1,2
=3,4
4021
ISFLG=ISFLG +2
XA(IMAX+2-IY)= XA(IMAX+1-IY)
YA(IMAX+2-IY)= YA(IMAX+1-IY)
403
ZA(IMAX+2-IY)= ZA(IMAX+1-IY)
XL=XA(NS)=XAI
YL=YA(NS)=YAI
CALL ELEV
ZA(NS)=ZE
IMAX=IMAX+1
407
IPP=0
ISFLG
=1,2
=3,4
404
ISFLG=ISFLG +1
922
P
P 65
IF IPP-I9
<0
≥0
408
IPP=0



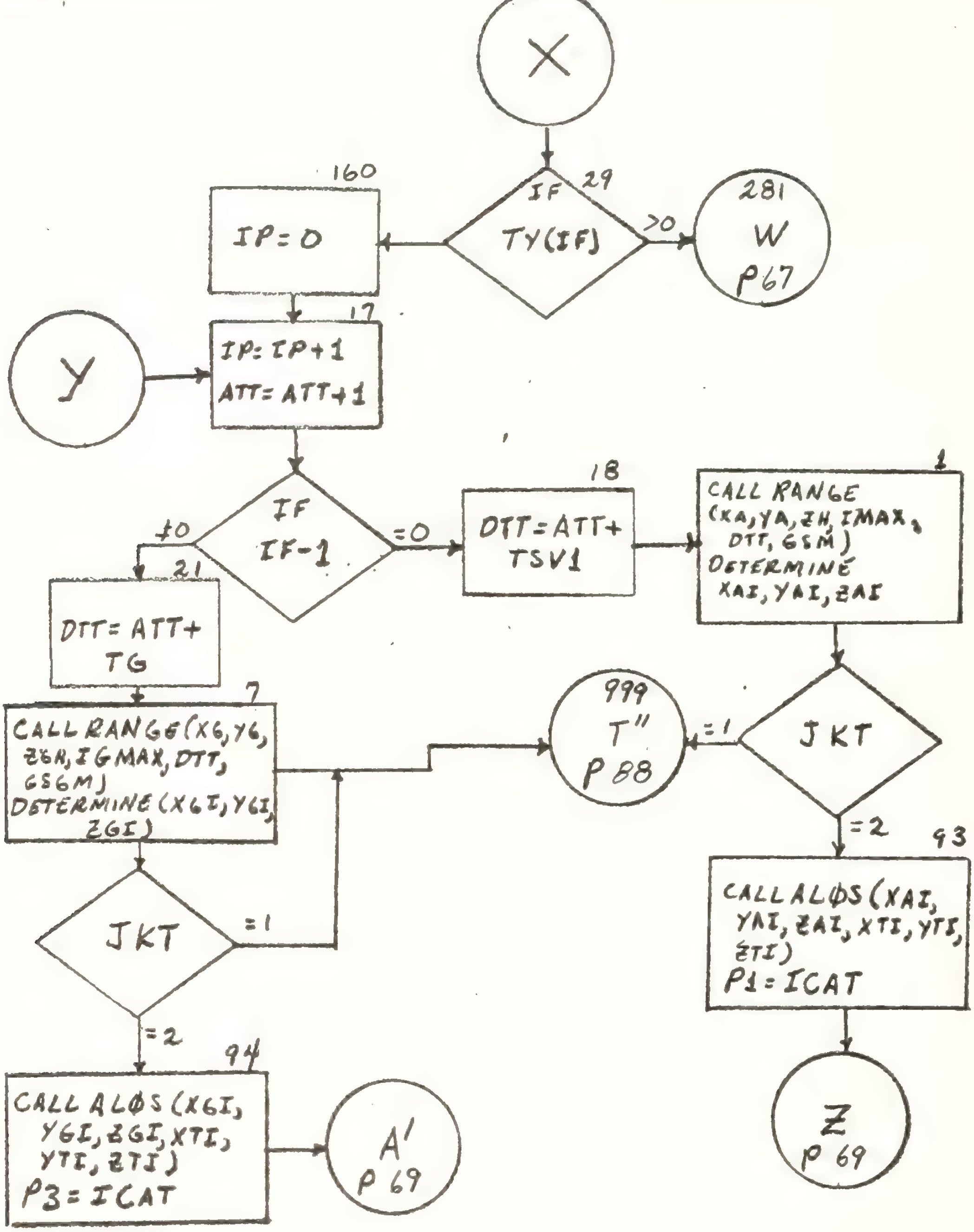
SUBROUTINE ATTACK (INITIAL A/C DECISIONS)
PAGE 10 OF 33
P
922
IF IGO-1
924
P4=1
IGFLG=2
XGI=YGI=ZGI=0
>0
≤0
923
DTT=ATT+TG
CALL RANGE (ATTACK POSIT.)
CALCULATE (XGI, YGI, ZGI)
CALL ALØS(XGI, YGI, ZGI, XTI, YTI, ZTI)
P3=ICAT & P6=DIS
P4=ITSE & N6=N+1
IF P4-2
<0
2345
EF=ATT
≥0
3066
IF INSE-2
3064
A
P 56
925
PRINT ATTACK CONDITION
JKT
=2
=1
92
Q
P 66
999
"T"
P 88

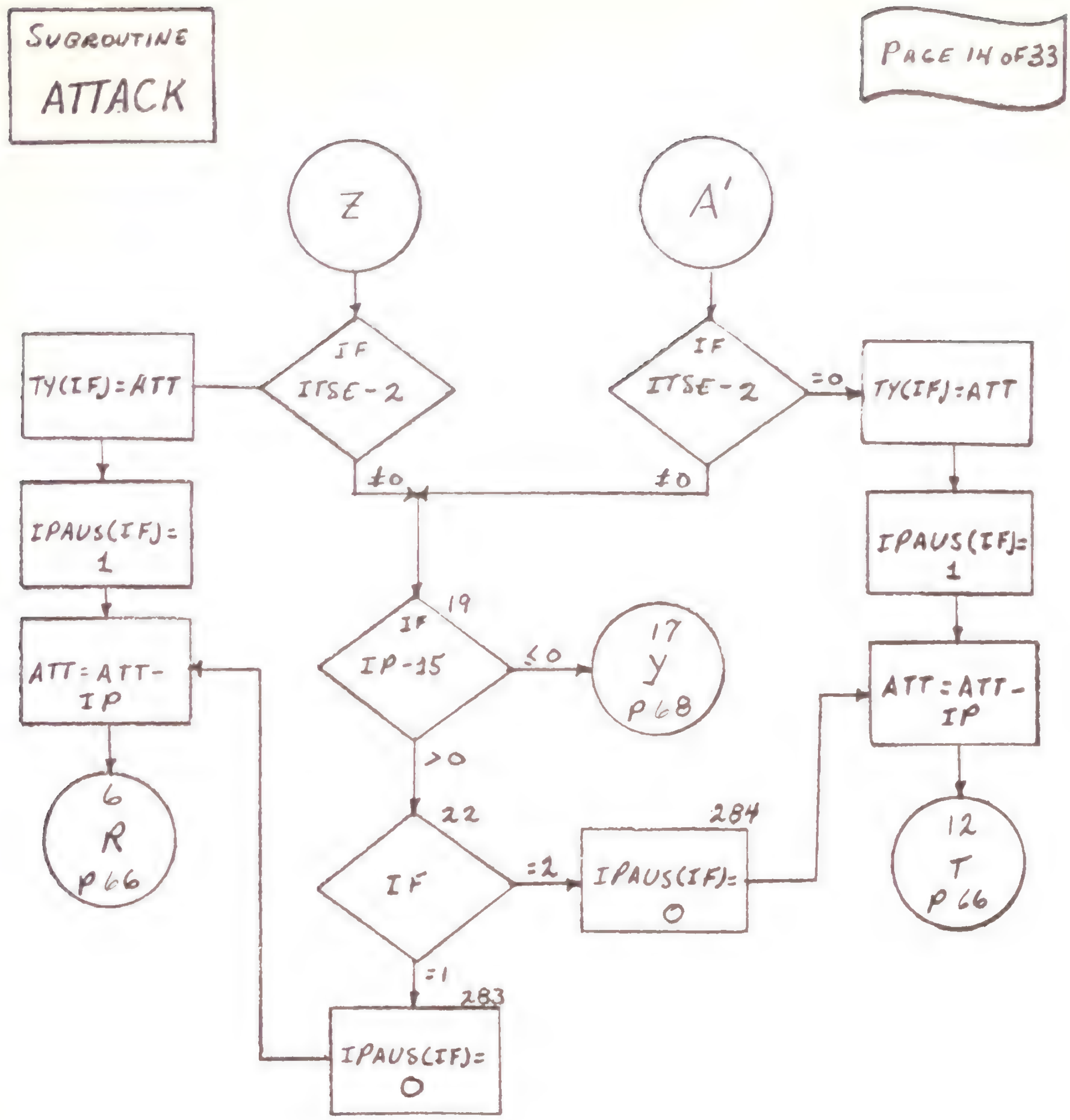

SUBROUTINE
ATTACK
(TARGET FIRE
CONTROL)
R
Q
9
IF
RANF(-1.)-SPT(P1)
>0
≤0
2
ISFLG
=1,4
=0
92
IF
P2-2
≠0
20
V
P67
10
ISFLG=ISFLG
-1
AB=ATT
=3,3
3
IF
IPAUS(1)
≤0
>0
704
IPAUS(1)=0
AB=ATT
IF
IA-2*NWPTYP
>0
≤0
30
B'
P70
647
ITMØV=1
S
11
U
P67
6
ID=1
703
ID=3
TY(1)=0
90
IF
P4-2
≠0
=0
705
W"
P70
T
14
IF
RANF(-1.)-SPT(P3)
>0
≤0
=1,4
4
IGFLG
=3,3
270
IF
IPAUS(2)
≤0
>0
706
X"
P70
12
ID=ID+1
30
B'
P70
140
IGFLG=IGFLG
-1
IF
IA-2*NWPTYP
≤0
>0
648
ITMØV=1


SUBROUTINE
ATTACK
PAGE 12 OF 33
U
V
11
IGFLG
=2,4
12
T
P66
=1,3
13
IF=2
20
ISFLG
=2,4
6
R
P 65
=1,3
8
IF=1
16
IF
ATT-TY(IF)
>0
29
X
P 68
≤0
IF
981
IF-1
≤0
>0
281
IPAUS(IF)=
0
W

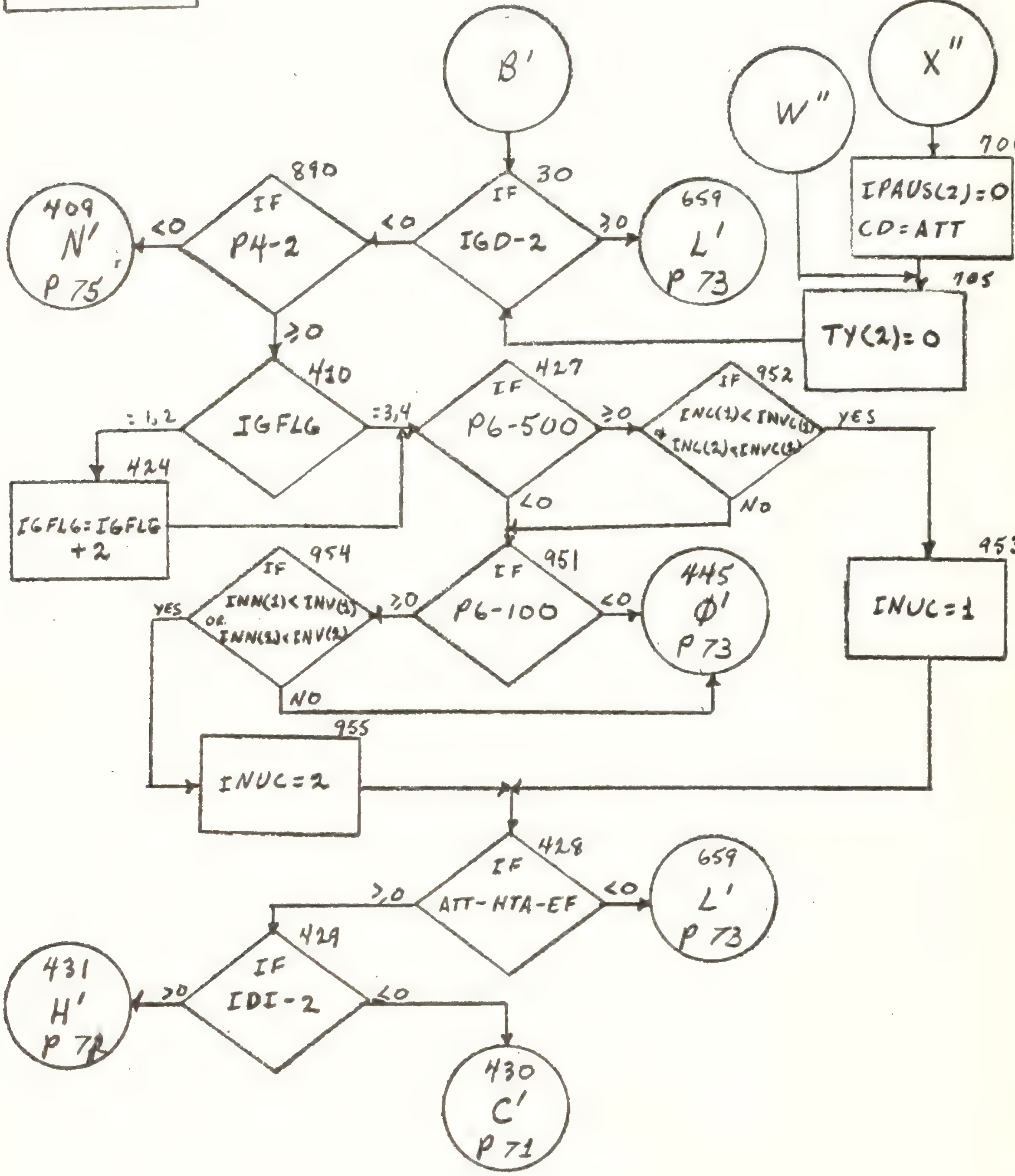
SUBROUTINE

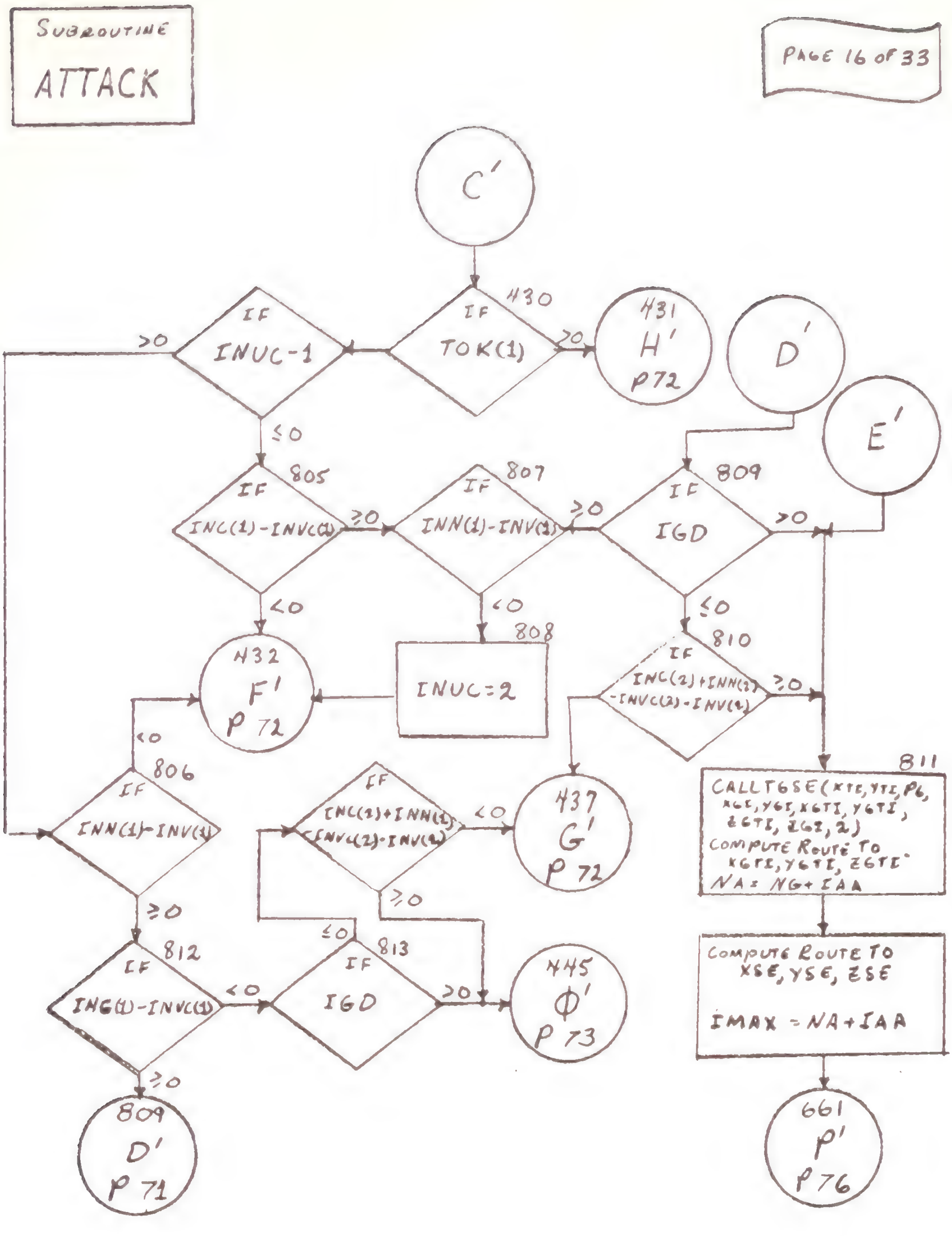
ATTACK




SUBROUTINE
ATTACK
PAGE 14 OF 33
Z
A'
IF
ITSE-2
IF
ITSE-2
=0
TY(IF)=ATT
TY(IF)=ATT
≠0
≠0
IPAUS(IF)=
1
IPAUS(IF)=
1
19
IF
IP-15
≤0
17
Y
P68
ATT=ATT-
IP
ATT=ATT-
IP
>0
6
R
P66
22
IF
=2
284
IPAUS(IF)=
0
12
T
P66
=1
283
IPAUS(IF)=
0

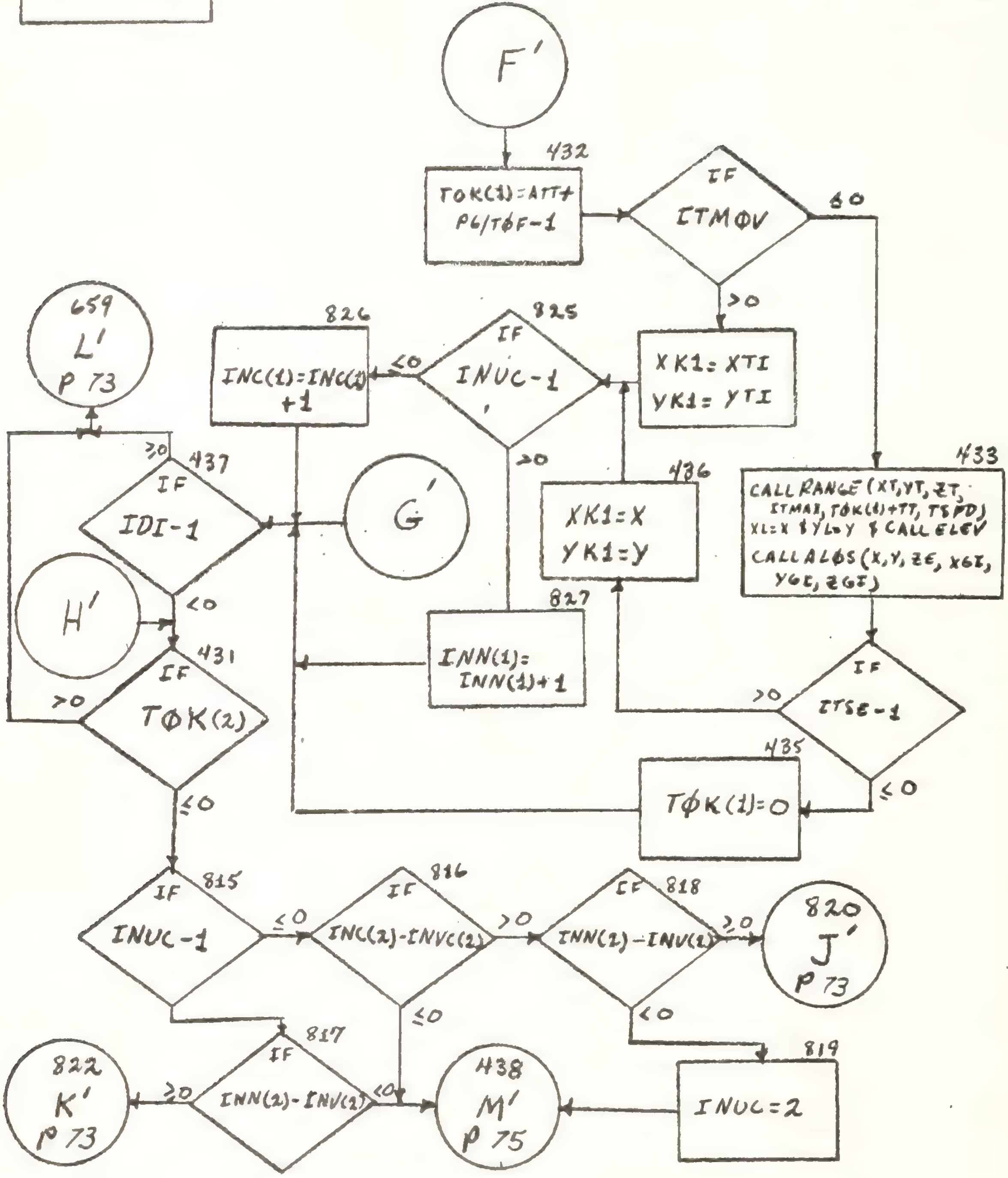
SUBROUTINE
ATTACK
(A/C ATTACK
& FIRE CONTROL)

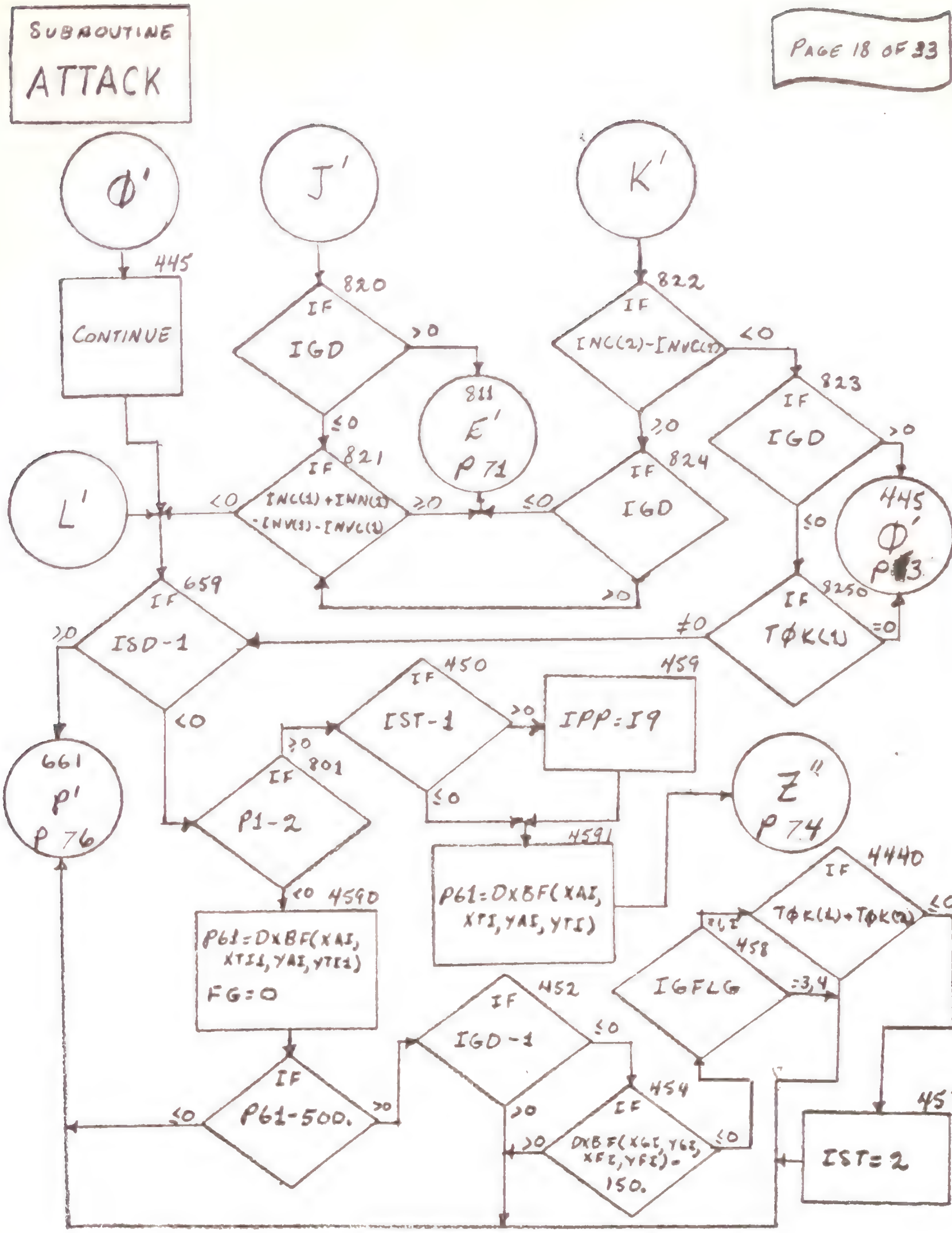




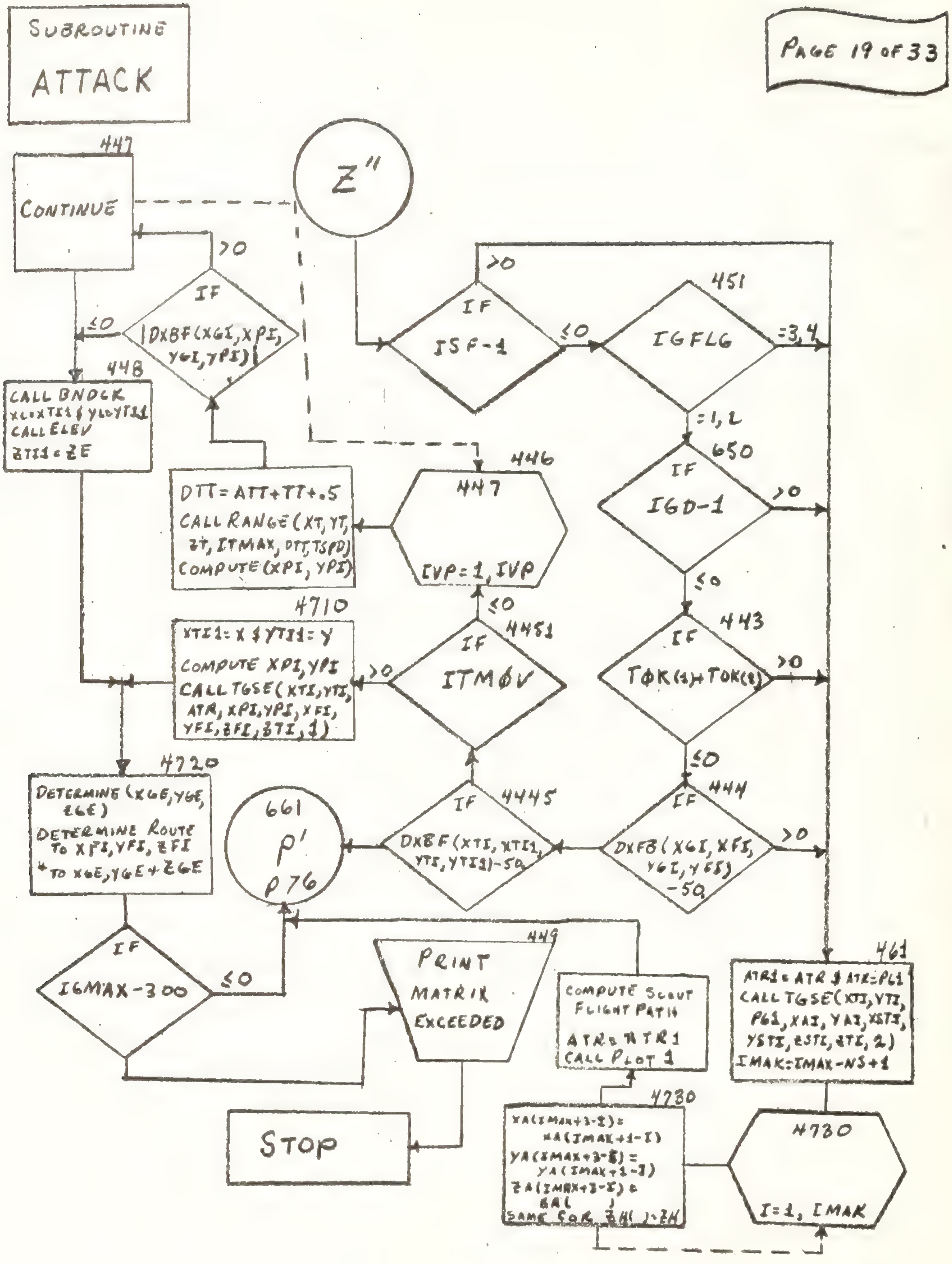
SUBROUTINE
ATTACK
PAGE 16 OF 33
C'
430
IF
TOK(1)
>0
431
H'
P72
IF
INUC-1
>0
≤0
D'
E'
805
IF
INC(1)-INVC(1)
≥0
<0
807
IF
INN(1)-INV(1)
<0
809
IF
IGD
>0
≤0
808
INUC=2
432
F'
P 72
810
IF
INC(2)+INN(2)
-INVC(2)-INV(2)
≥0
811
CALLTGSE(XTE,YTE,PG,
XGE,YGE,XGTI,YGTI,
ZGTI,ZGE,2)
COMPUTE ROUTE TO
XGTI,YGTI,ZGTI
NA=NG+IAA
806
IF
INN(1)-INV(1)
<0
≥0
IF
INC(2)+INN(2)
-INVC(2)-INV(2)
<0
≥0
437
G'
P 72
COMPUTE ROUTE TO
XSE, YSE, ZSE
IMAX = NA+IAA
812
IF
ING(1)-INVC(1)
<0
≥0
813
IF
IGD
≤0
>0
445
Φ'
P 73
809
D'
P 71
661
P'
P76

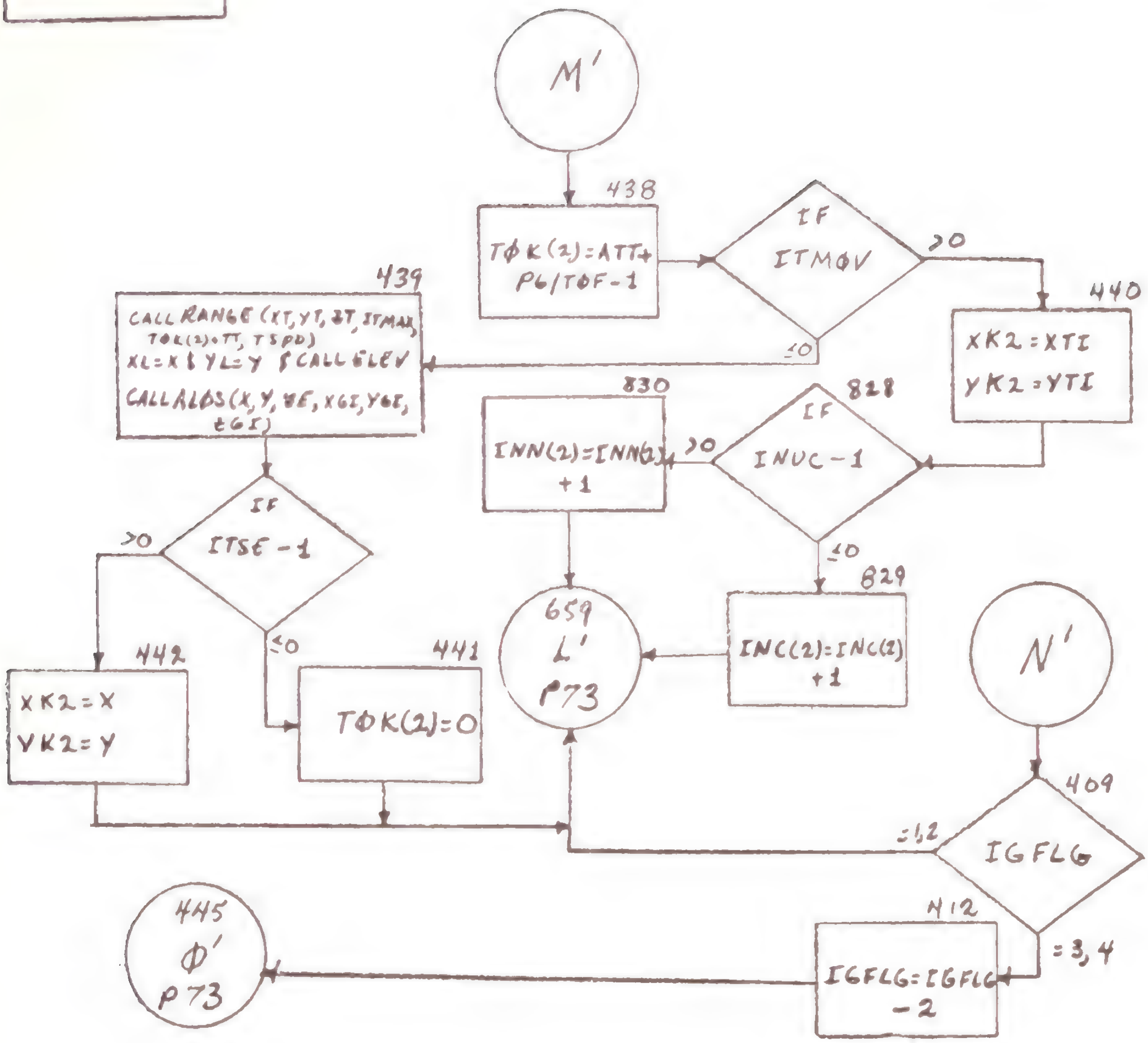
Subroutine ATTACK

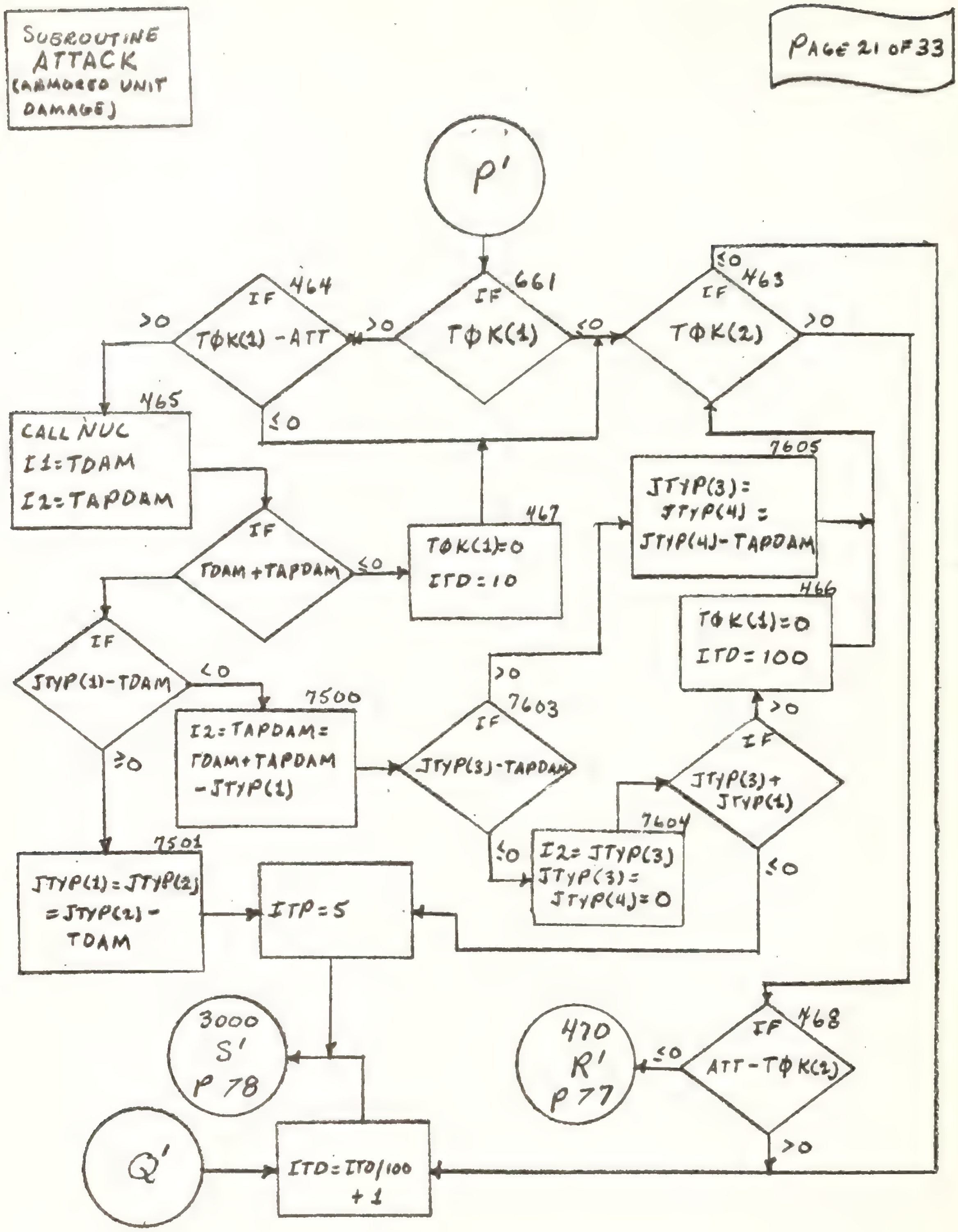




SUBROUTINE
ATTACK
PAGE 18 OF 33
Φ'
J'
K'
445
CONTINUE
820
IF
IGD
>0
≤0
811
E'
P 71
821
IF
INC(1)+INN(1)
-INV(1)-INVC(1)
<0
≥0
822
IF
INC(2)-INVC(2)
<0
≥0
823
IF
IGD
>0
≤0
824
IF
IGD
≤0
>0
445
Φ'
L'
659
IF
ISD-1
≥0
<0
9250
IF
TΦK(1)
≠0
=0
661
P'
P 76
450
IF
IST-1
>0
≤0
459
IPP=I9
801
IF
P1-2
≥0
<0
Z''
P 74
4591
P61=DXBF(XAI,
XTI,YAI,YTI)
4590
P61=DXBF(XAI,
XTI1,YAI,YTI1)
FG=0
4440
IF
TΦK(1)+TΦK(2)
≤0
458
IGFLG
=1,2
=3,4
452
IF
IGD-1
≤0
>0
IF
P61-500.
≤0
>0
454
IF
DXBF(XGI,YGI,
XFI,YFI)-
150.
>0
≤0
457
IST=2


SUBROUTINE
ATTACK
447
CONTINUE
Z''
>0
IF
|DXBF(XGI, XPI, YGI, YPI)|
≤0
448
CALL BNDCK
XL=XTI1 $ YL=YTI1
CALL ELEV
ZTI1=ZE
DTT=ATT+TT+.5
CALL RANGE(XT, YT, ZT, ITMAX, DTT, TSPD)
COMPUTE(XPI, YPI)
446
447
IVP=1, IVP
>0
IF
ISF-1
≤0
451
IGFLG
=3,4
=1,2
650
IF
IGD-1
>0
≤0
443
IF
TØK(1)+TOK(2)
>0
≤0
444
IF
DXFB(XGI, XFI, YGI, YFI) -50
>0
4710
XTI1=X $ YTI1=Y
COMPUTE XPI, YPI
CALL TGSE(XTI, YTI, ATR, XPI, YPI, XFI, YFI, ZFI, ZTI, 1)
>0
4451
IF
ITMØV
≤0
4720
DETERMINE (XGE, YGE, ZGE)
DETERMINE ROUTE TO XFI, YFI, ZFI + TO XGE, YGE + ZGE
661
ρ'
ρ76
4445
IF
DXBF(XTI, XTI1, YTI, YTI1)-50
IF
IGMAX-300
≤0
449
PRINT
MATRIX
EXCEEDED
STOP
COMPUTE SCOUT FLIGHT PATH
ATR=ATR1
CALL PLOT 1
461
ATR1=ATR $ ATR=PL1
CALL TGSE(XTI, YTI, PG1, XAI, YAI, XSTI, YSTI, ZSTI, ZTI, 2)
IMAX=IMAX-NS+1
4730
XA(IMAX+3-I)= XA(IMAX+1-I)
YA(IMAX+3-I)= YA(IMAX+1-I)
ZA(IMAX+3-I)= ZA( )
SAME FOR ZH( )-ZH
4730
I=1, IMAX

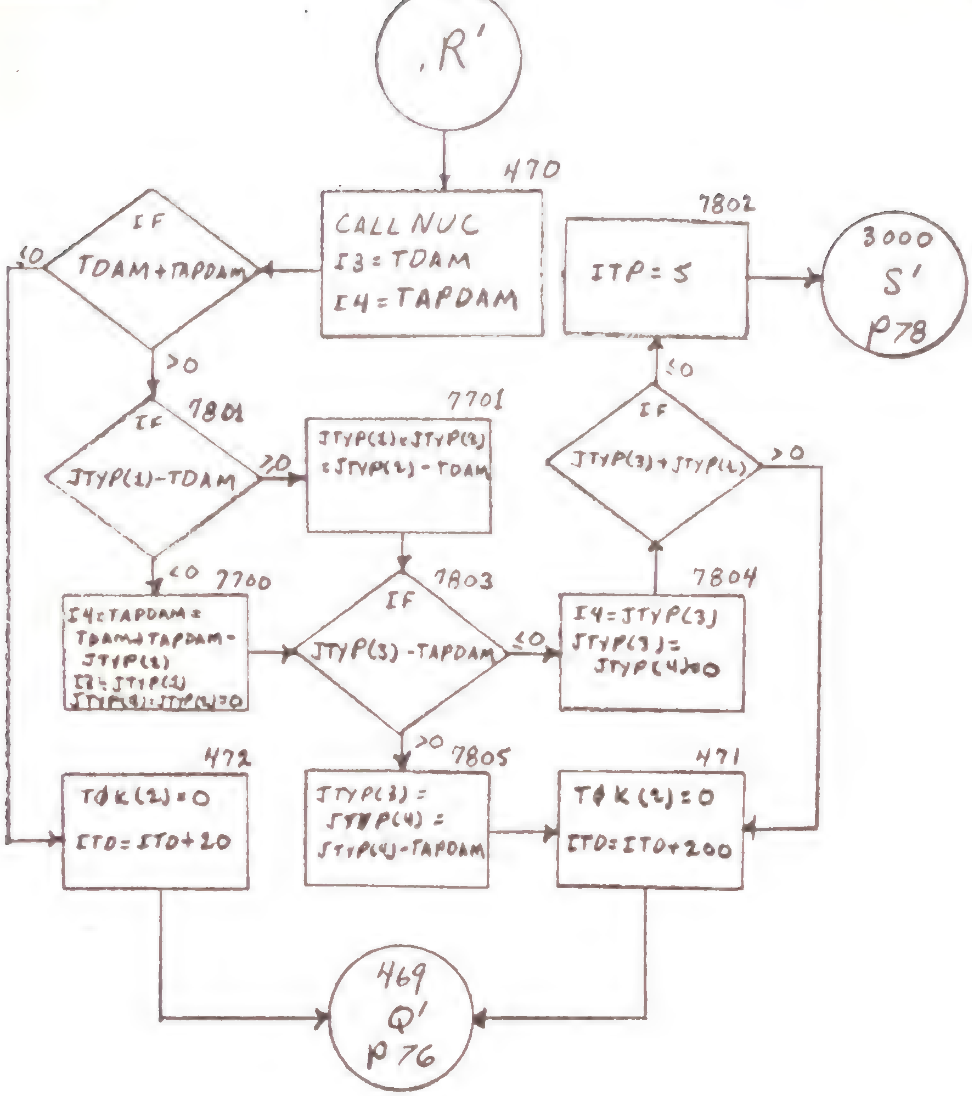

M'
438
TØK(2)=ATT+P6/TOF-1
IF ITMØV
>0
≤0
440
XK2=XTI
YK2=YTI
439
CALL RANGE (XT, YT, ZT, ITMAX, TOK(2)+TT, TSPD)
XL=X $ YL=Y $ CALL ELEV
CALL ALOS (X, Y, ZE, XGI, YGI, ZGI)
830
INN(2)=INN(2)+1
828
IF INUC-1
>0
≤0
829
INC(2)=INC(2)+1
IF ITSE-1
>0
≤0
442
XK2=X
YK2=Y
441
TØK(2)=0
659
L'
P73
N'
409
IGFLG
=1,2
=3,4
412
IGFLG=IGFLG-2
445
Φ'
P73

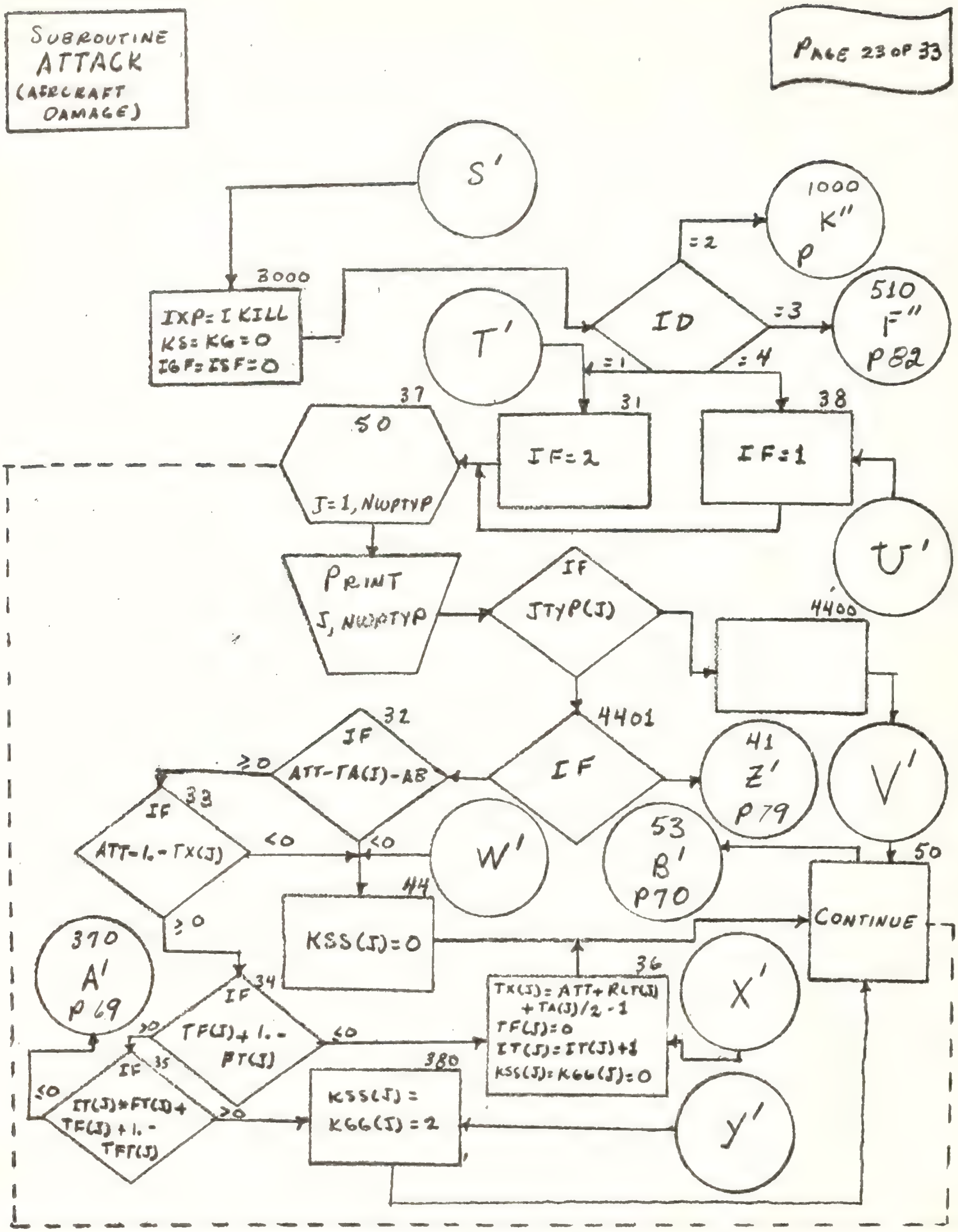

SUBROUTINE ATTACK (ARMORED UNIT DAMAGE)
PAGE 21 OF 33
P'
IF 464 TΦK(1) - ATT
>0
≤0
IF 661 TΦK(1)
>0
≤0
IF 463 TΦK(2)
≤0
>0
465
CALL NUC
I1=TDAM
I2=TAPDAM
IF TDAM+TAPDAM
≤0
467
TΦK(1)=0
ITD=10
7605
JTYP(3)=
JTYP(4)=
JTYP(4)-TAPDAM
466
TΦK(1)=0
ITD=100
IF JTYP(1)-TDAM
<0
≥0
7500
I2=TAPDAM=
TDAM+TAPDAM
-JTYP(1)
IF 7603 JTYP(3)-TAPDAM
>0
≤0
IF JTYP(3)+ JTYP(1)
>0
≤0
7604
I2=JTYP(3)
JTYP(3)=
JTYP(4)=0
7501
JTYP(1)=JTYP(2)
=JTYP(2)-
TDAM
ITP=5
3000
S'
P 78
IF 468 ATT-TΦK(2)
≤0
>0
470
R'
P 77
Q'
ITD=ITD/100
+1

SUBROUTINE

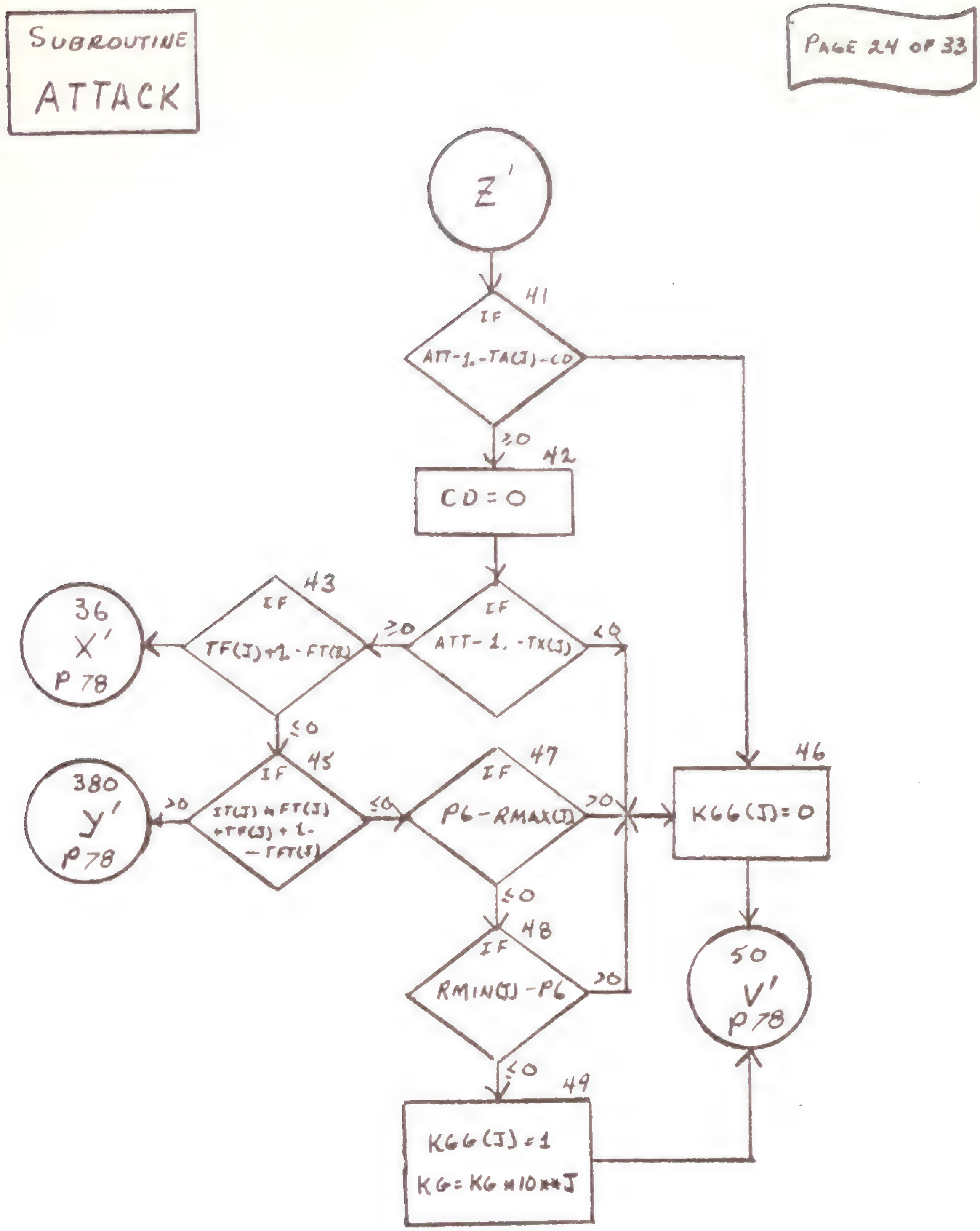
ATTACK

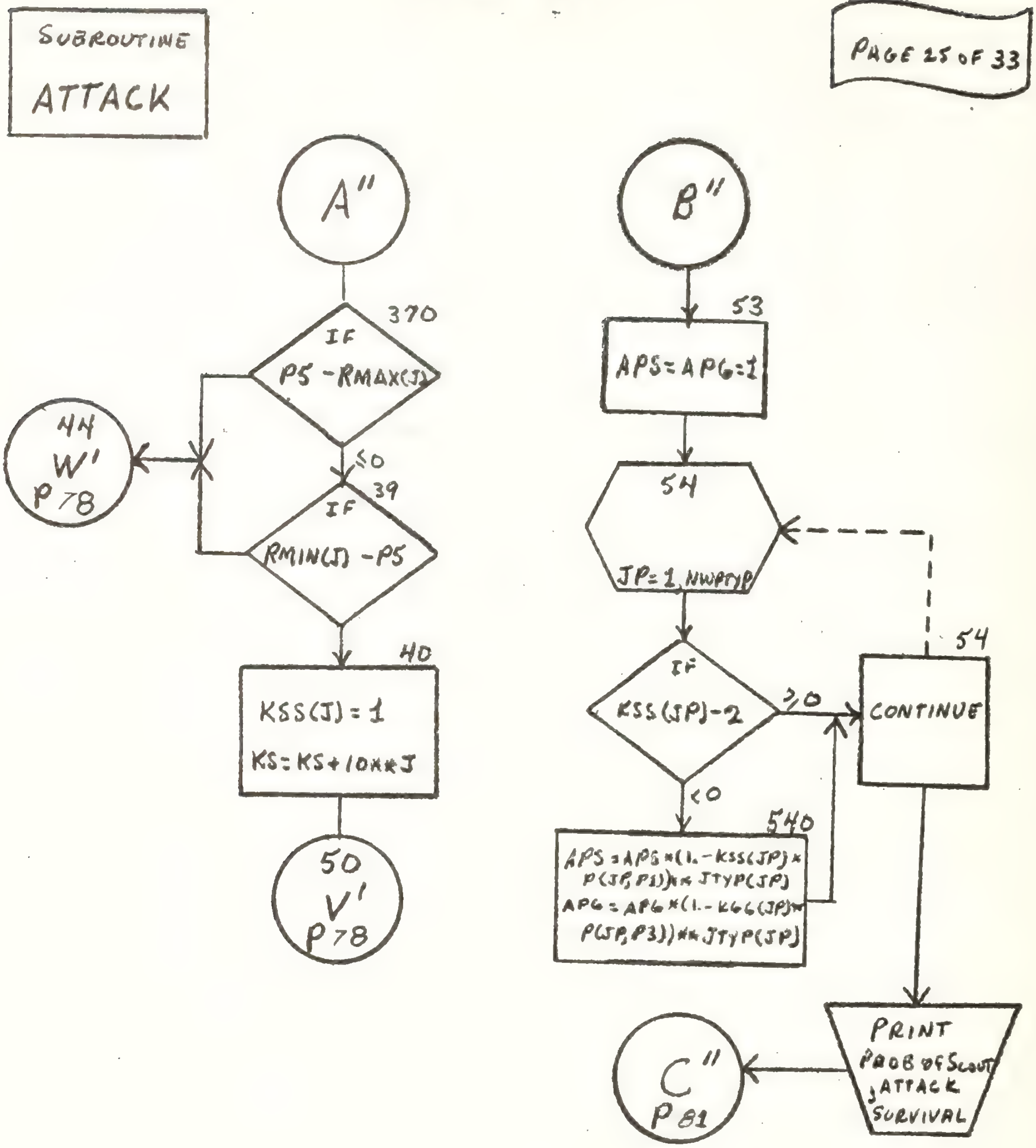
PAGE 22 OF 33

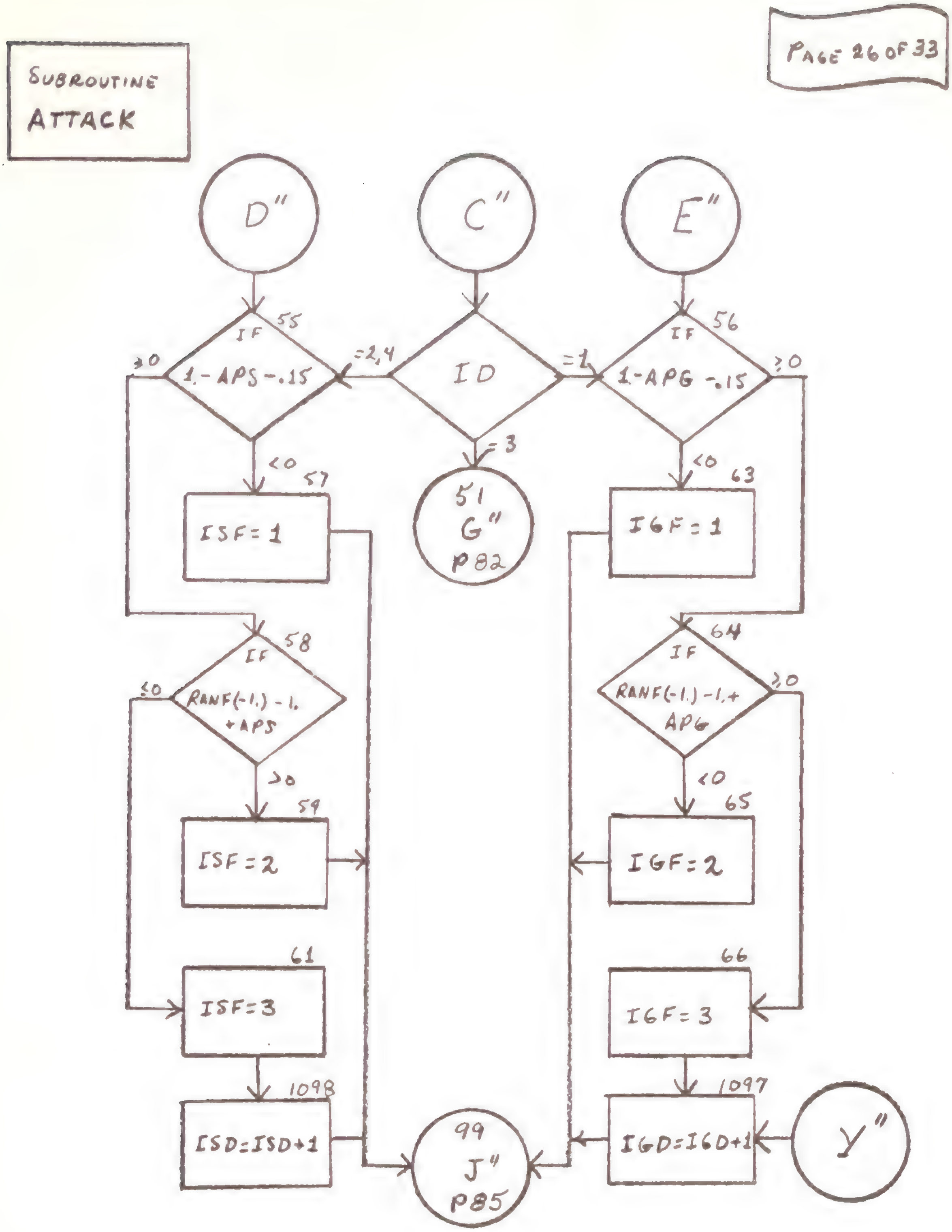




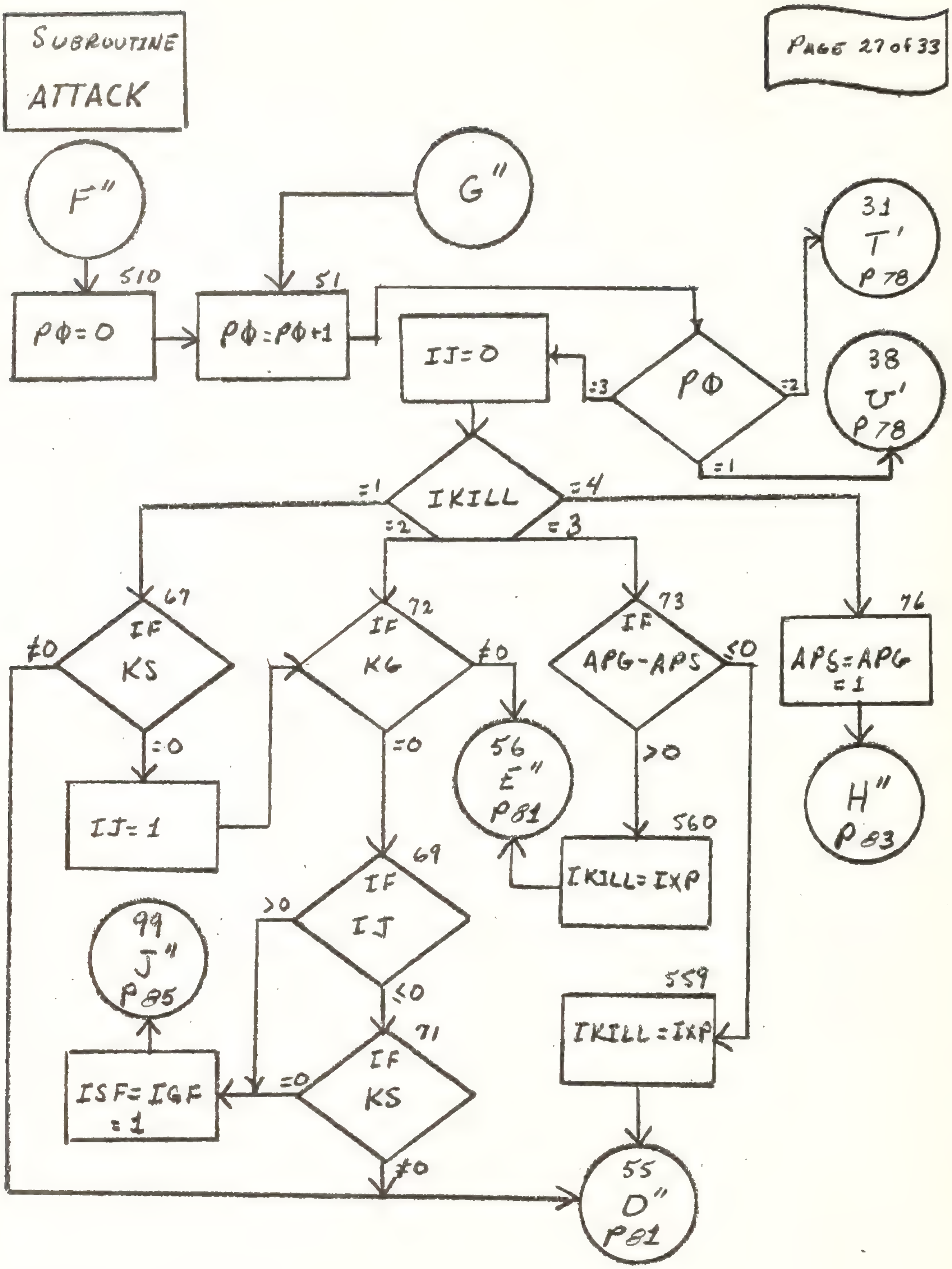
SUBROUTINE ATTACK (AIRCRAFT DAMAGE)
PAGE 23 OF 33
S'
3000
IXP= I KILL
KS= KG = 0
IGF= ISF= 0
ID
=2
1000
K''
P
=3
510
F''
P82
=1
=4
T'
31
IF=2
38
IF=1
37
50
J=1, NWPTYP
U'
PRINT
J, NWPTYP
IF
JTYP(J)
4400
4401
IF
41
Z'
P79
V'
32
IF
ATT-TA(J)-AB
≥0
<0
33
IF
ATT-1.-TX(J)
W'
53
B'
P70
50
CONTINUE
44
KSS(J)=0
≥0
370
A'
P69
34
IF
TF(J)+1.-FT(J)
<0
>0
36
TX(J)= ATT+RLT(J) + TA(J)/2 - 1
TF(J)=0
IT(J)=IT(J)+1
KSS(J)=KGG(J)=0
X'
35
IF
IT(J)*FT(J)+TF(J)+1.-TFT(J)
≤0
>0
380
KSS(J) = KGG(J) = 2
Y'


SUBROUTINE
ATTACK
PAGE 24 OF 33
Z'
41
IF
ATT-1.-TA(J)-CD
≥0
42
CD=0
43
IF
TF(J)+1.-FT(J)
>0
IF
ATT-1.-TX(J)
<0
36
X'
P 78
≤0
45
IF
ST(J)*FT(J)
+TF(J)+1.
-TFT(J)
>0
380
Y'
P 78
≤0
47
IF
P6-RMAX(J)
>0
46
K66(J)=0
≤0
48
IF
RMIN(J)-P6
>0
50
V'
P 78
≤0
49
K66(J)=1
KG=KG*10**J


SUBROUTINE
ATTACK
PAGE 25 OF 33
A''
370
IF
P5 -RMAX(J)
44
W'
P78
≤0
39
IF
RMIN(J) -P5
40
KSS(J) = 1
KS=KS+10**J
50
V'
P78
B''
53
APS=AP6=1
54
JP=1,NWPTYP
IF
KSS(JP)-2
≥0
<0
54
CONTINUE
540
APS=APS*(1.-KSS(JP)*P(JP,P1))**JTYP(JP)
AP6=AP6*(1.-K66(JP)*P(JP,P3))**JTYP(JP)
PRINT
PROB OF SCOUT ATTACK SURVIVAL
C''
P81

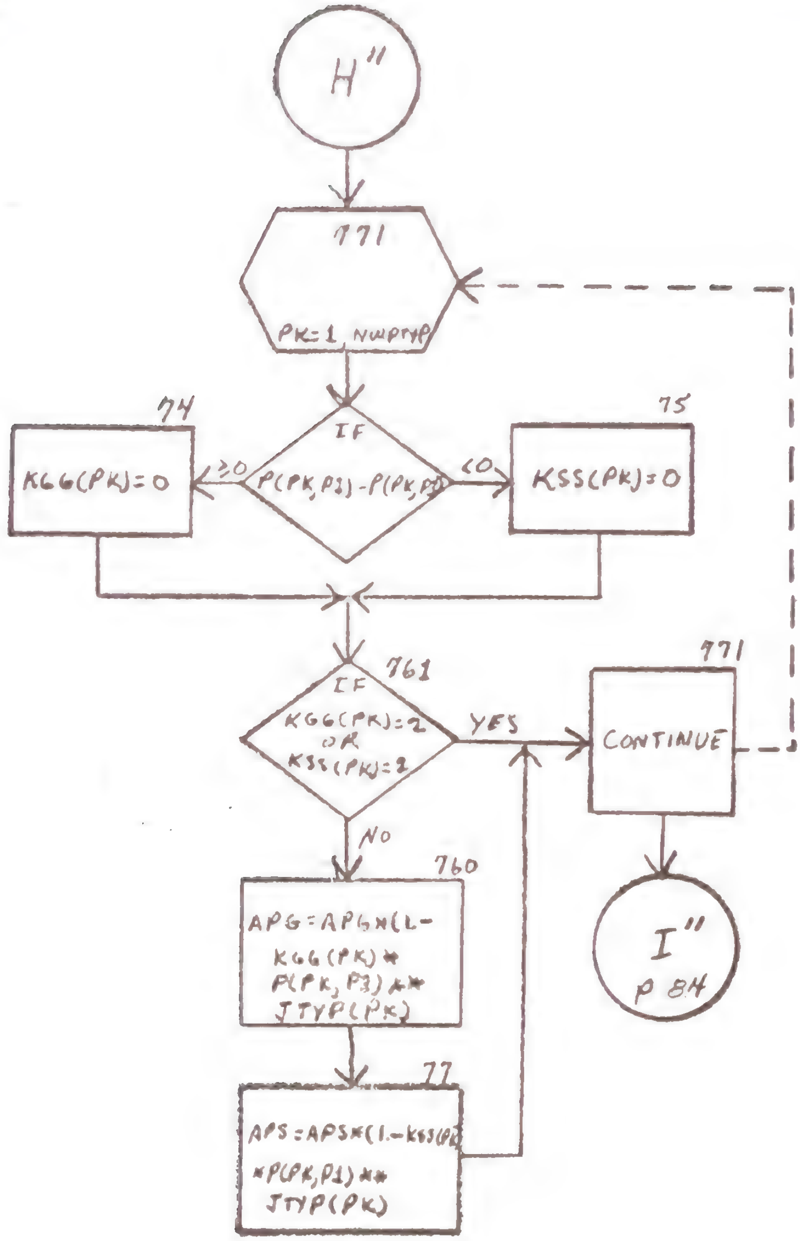


SUBROUTINE ATTACK
PAGE 26 OF 33
D''
C''
E''
55
IF
1.-APS-.15
≥0
<0
=2,4
ID
=1
=3
56
IF
1-APG-.15
≥0
<0
57
ISF=1
51
G''
P82
63
IGF=1
58
IF
RANF(-1.)-1.+APS
≤0
≥0
64
IF
RANF(-1.)-1.+APG
≥0
<0
59
ISF=2
65
IGF=2
61
ISF=3
66
IGF=3
1098
ISD=ISD+1
99
J''
P85
1097
IGD=IGD+1
Y''



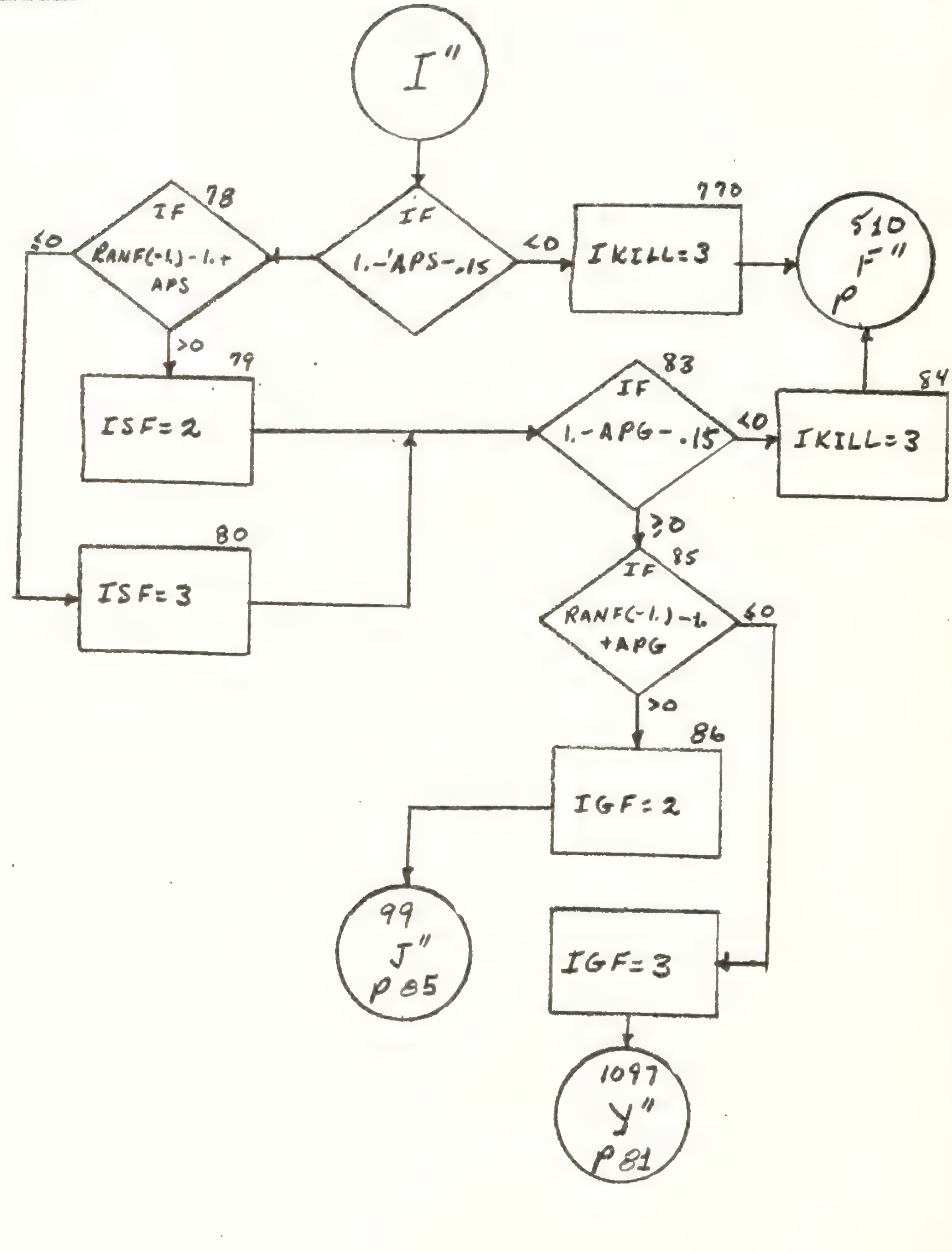
SUBROUTINE ATTACK
PAGE 27 of 33
F"
G"
510
PΦ=0
51
PΦ=PΦ+1
IJ=0
PΦ
=3
=2
=1
31 T' P 78
38 U' P 78
IKILL
=1
=2
=3
=4
67
IF KS
≠0
=0
IJ=1
72
IF KG
≠0
=0
56 E" P81
73
IF APG-APS
≤0
>0
560
IKILL=IXP
76
APS=APG =1
H" P83
69
IF IJ
>0
≤0
71
IF KS
=0
≠0
99 J" P85
ISF=IGF =1
559
IKILL=IXP
55 D" P81

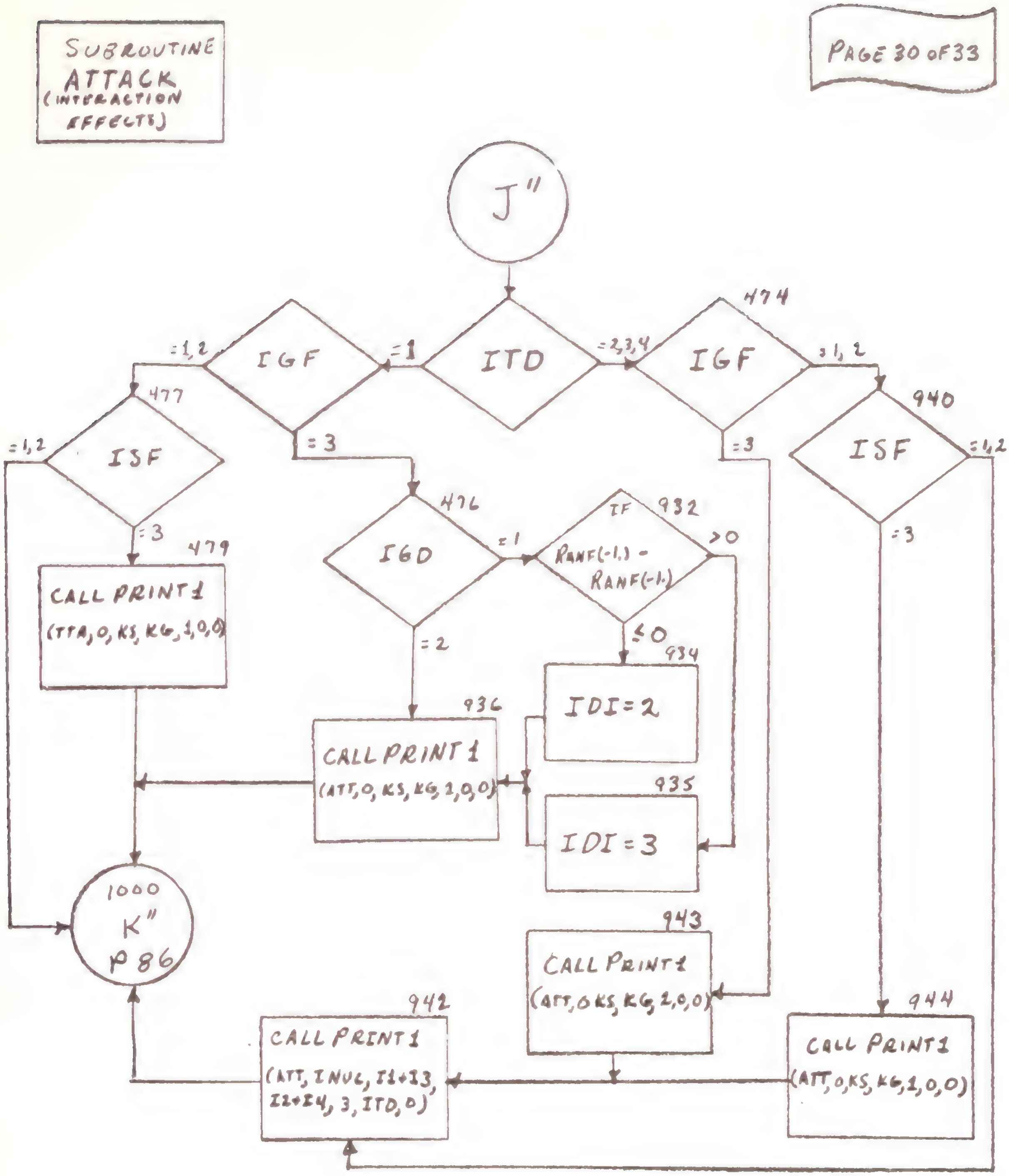
SUBROUTINE ATTACK

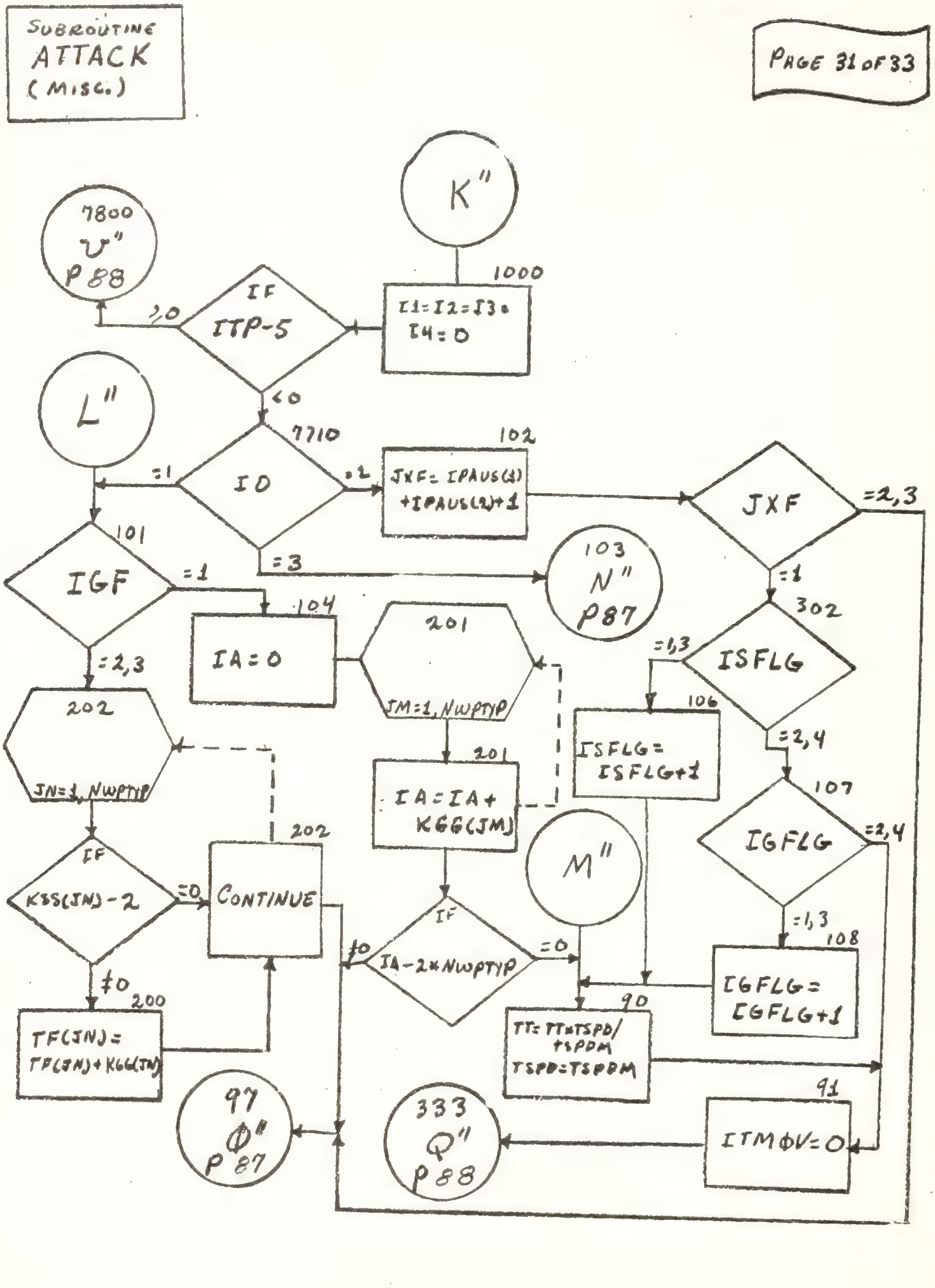


SUBROUTINE

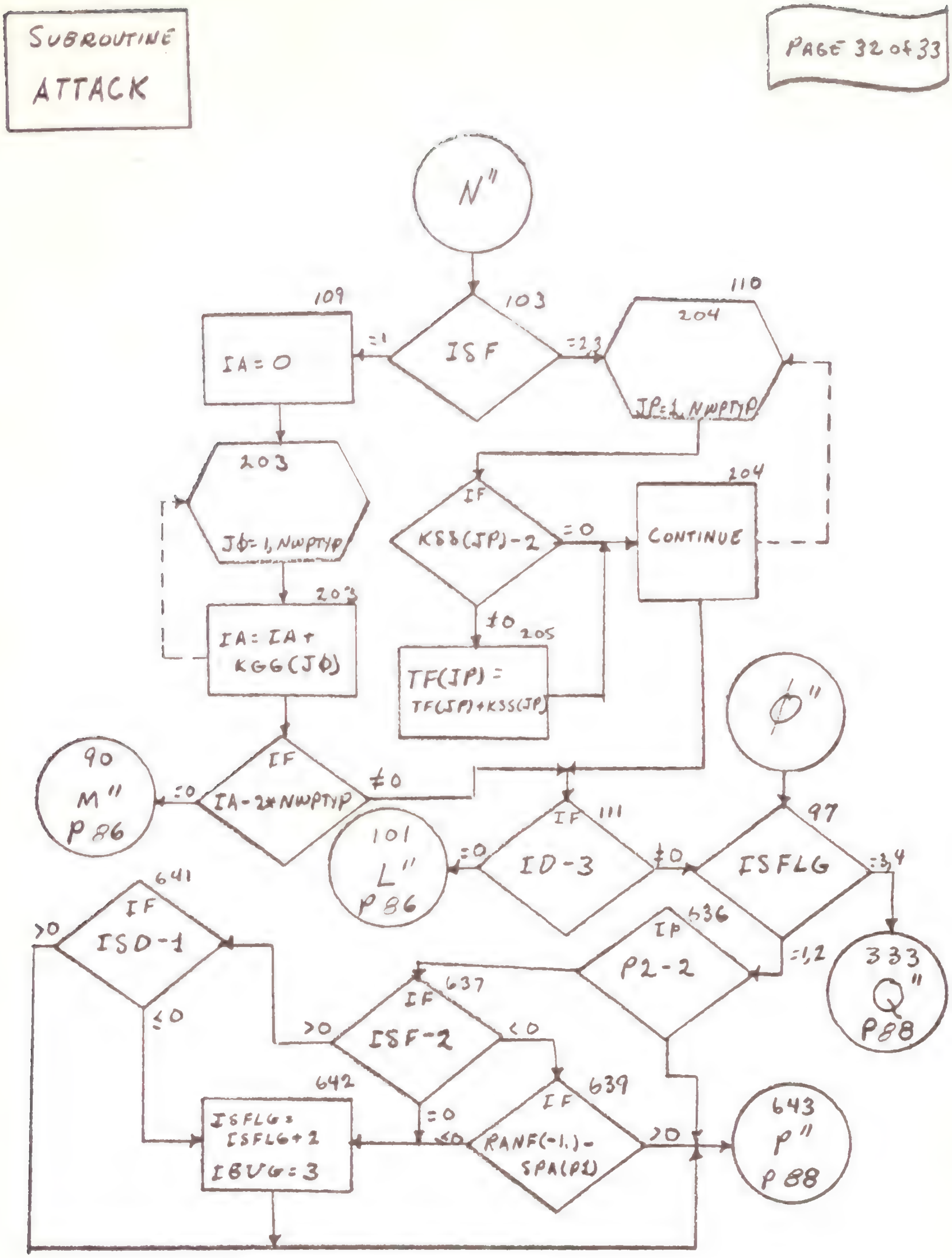
ATTACK

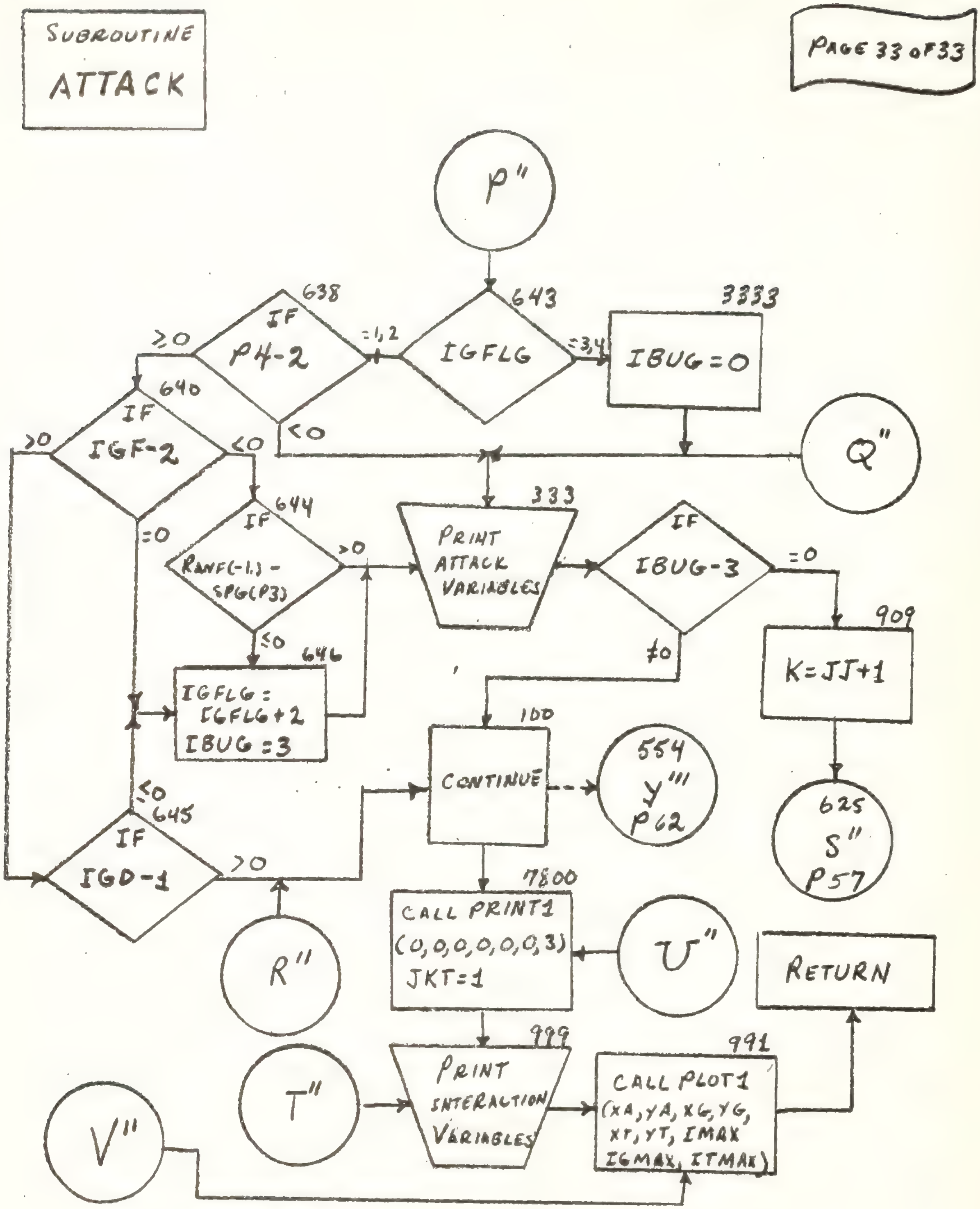


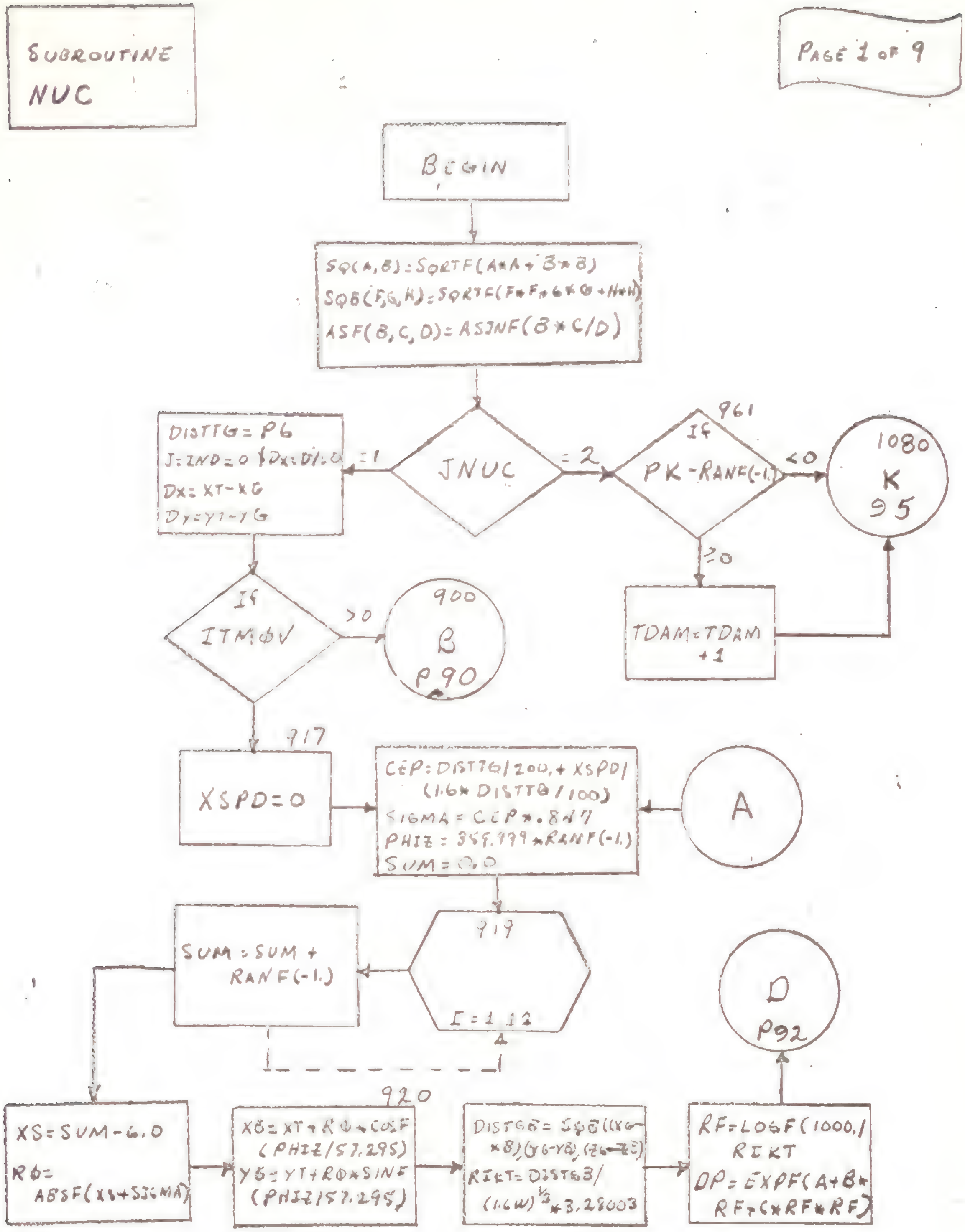

SUBROUTINE ATTACK (INTERACTION EFFECTS)
PAGE 30 OF 33
J"
ITD
=1
=2,3,4
IGF
=1,2
=3
474
IGF
=1,2
=3
477
ISF
=1,2
=3
940
ISF
=1,2
=3
479
CALL PRINT 1 (TTA, 0, KS, KG, 1, 0, 0)
476
IGD
=1
=2
932
IF RANF(-1.) - RANF(-1.)
>0
≤0
934
IDI=2
935
IDI=3
936
CALL PRINT 1 (ATT, 0, KS, KG, 2, 0, 0)
1000
K"
P 86
943
CALL PRINT 1 (ATT, 0 KS, KG, 2, 0, 0)
942
CALL PRINT 1 (ATT, INUC, I1+I3, I2+I4, 3, ITD, 0)
944
CALL PRINT 1 (ATT, 0, KS, KG, 1, 0, 0)



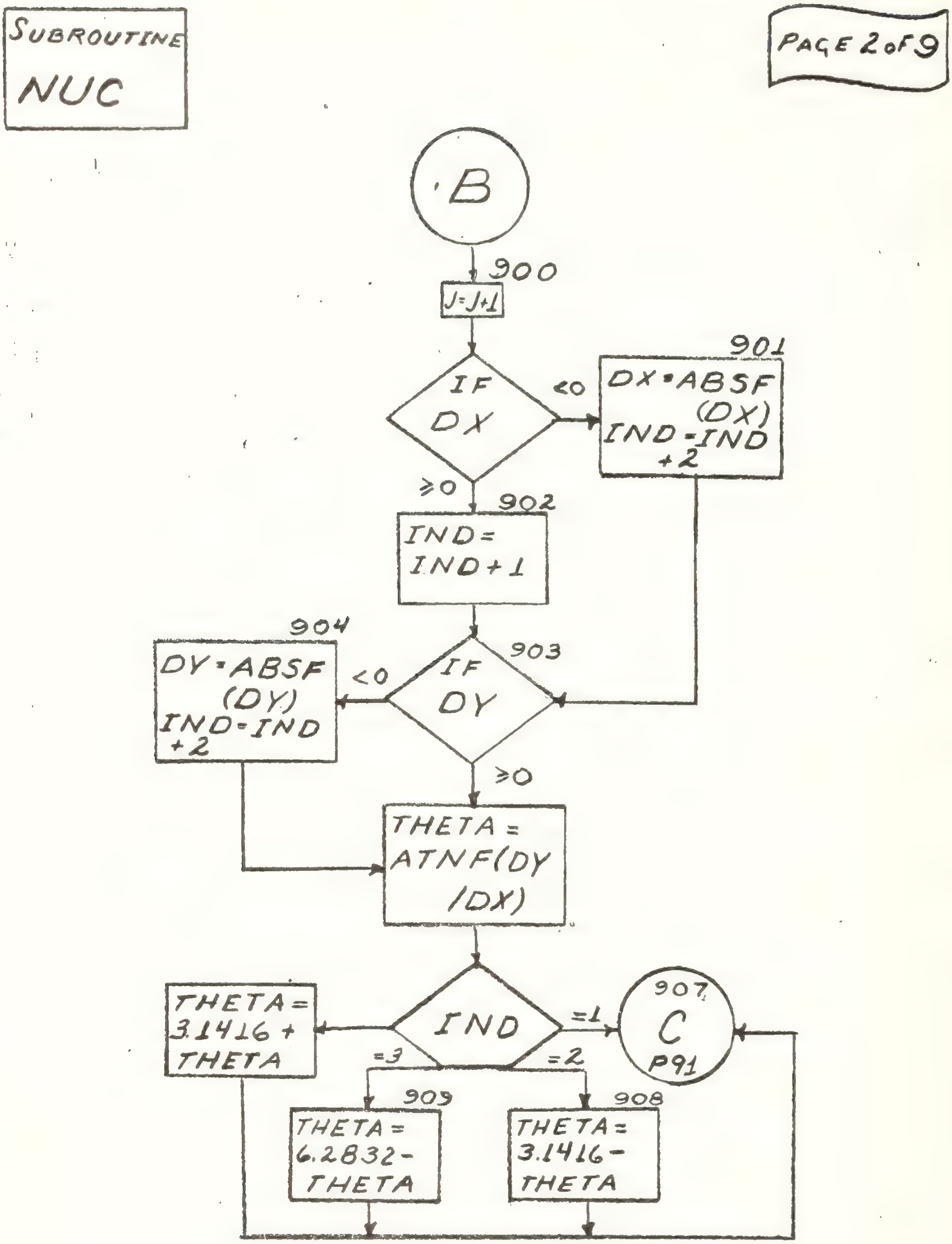
SUBROUTINE ATTACK (MISC.)
PAGE 31 OF 33
K"
7800 U" P88
1000
I1=I2=I3= I4=0
IF ITP-5
≥0
<0
L"
7710
ID
=1
=2
=3
102
JXF= IPAUS(1) +IPAUS(2)+1
JXF
=2,3
=1
101
IGF
=1
=2,3
103 N" P87
104
IA=0
201
JM=1, NWPTYP
201
IA=IA+ KGG(JM)
302
ISFLG
=1,3
=2,4
106
ISFLG= ISFLG+1
107
IGFLG
=2,4
=1,3
108
IGFLG= IGFLG+1
202
JN=1, NWPTYP
202
CONTINUE
IF KSS(JN)-2
=0
≠0
200
TF(JN)= TF(JN)+KGG(JN)
M"
IF IA-2*NWPTYP
≠0
=0
90
TT=TT*TSPD/ TSPDM TSPD=TSPDM
91
ITMΦV=0
97 Φ" P87
333 Q" P88

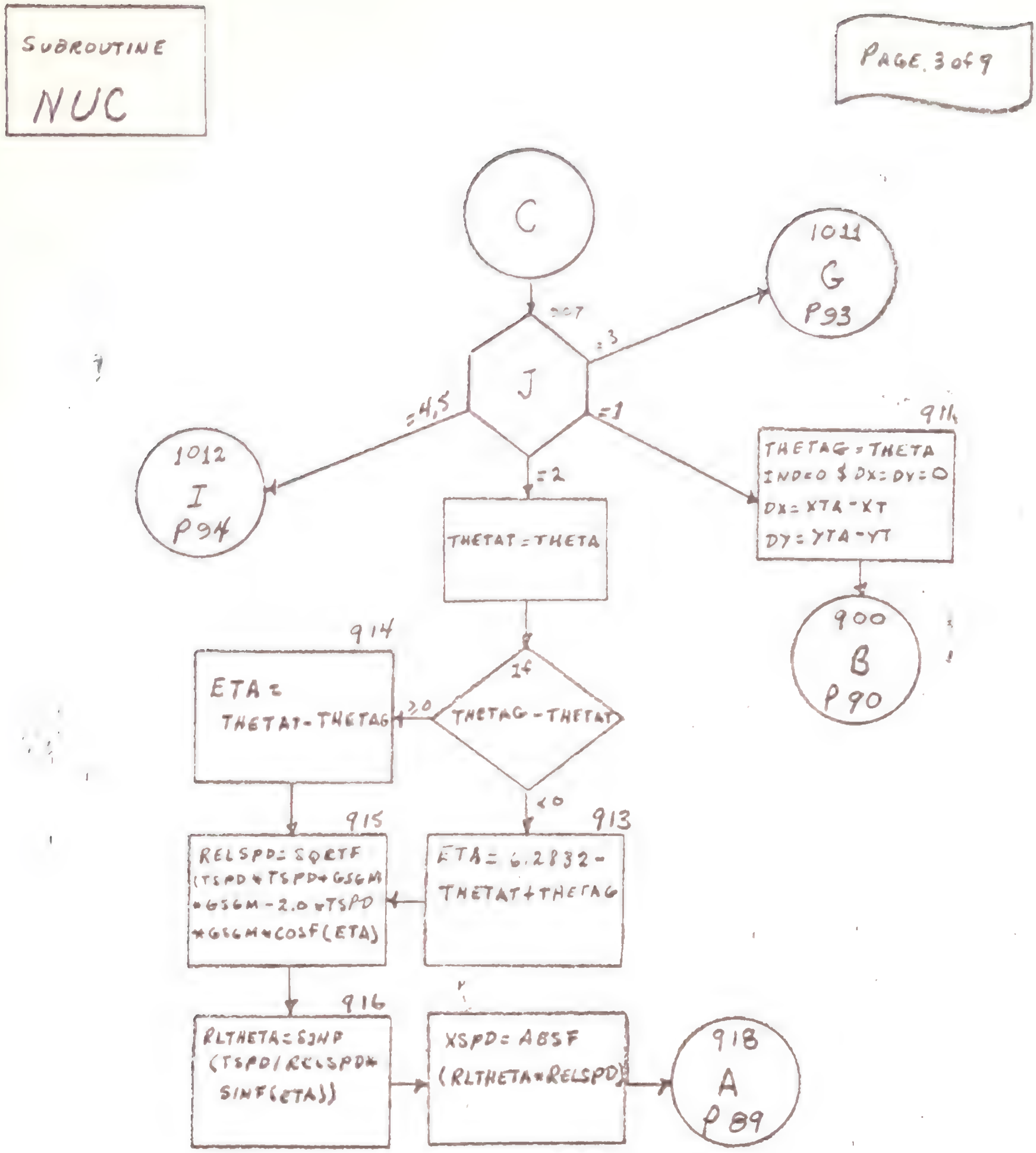

SUBROUTINE ATTACK
PAGE 32 of 33
N''
103
ISF
=1
=2,3
109
IA = 0
110
204
JP=1, NWPTYP
203
JΦ=1, NWPTYP
203
IA = IA + KGG(JΦ)
IF
KSS(JP) - 2
=0
≠0
204
CONTINUE
205
TF(JP) = TF(JP)+KSS(JP)
Φ''
IF
IA-2*NWPTYP
=0
≠0
90
M''
P 86
IF
ID-3
111
=0
≠0
101
L''
P 86
97
ISFLG
=3,4
=1,2
333
Q''
P88
641
IF
ISD-1
>0
≤0
636
IF
P2-2
637
IF
ISF-2
>0
<0
=0
639
IF
RANF(-1.)-SPA(P1)
≤0
>0
642
ISFLG = ISFLG+2
IBUG=3
643
P''
P 88

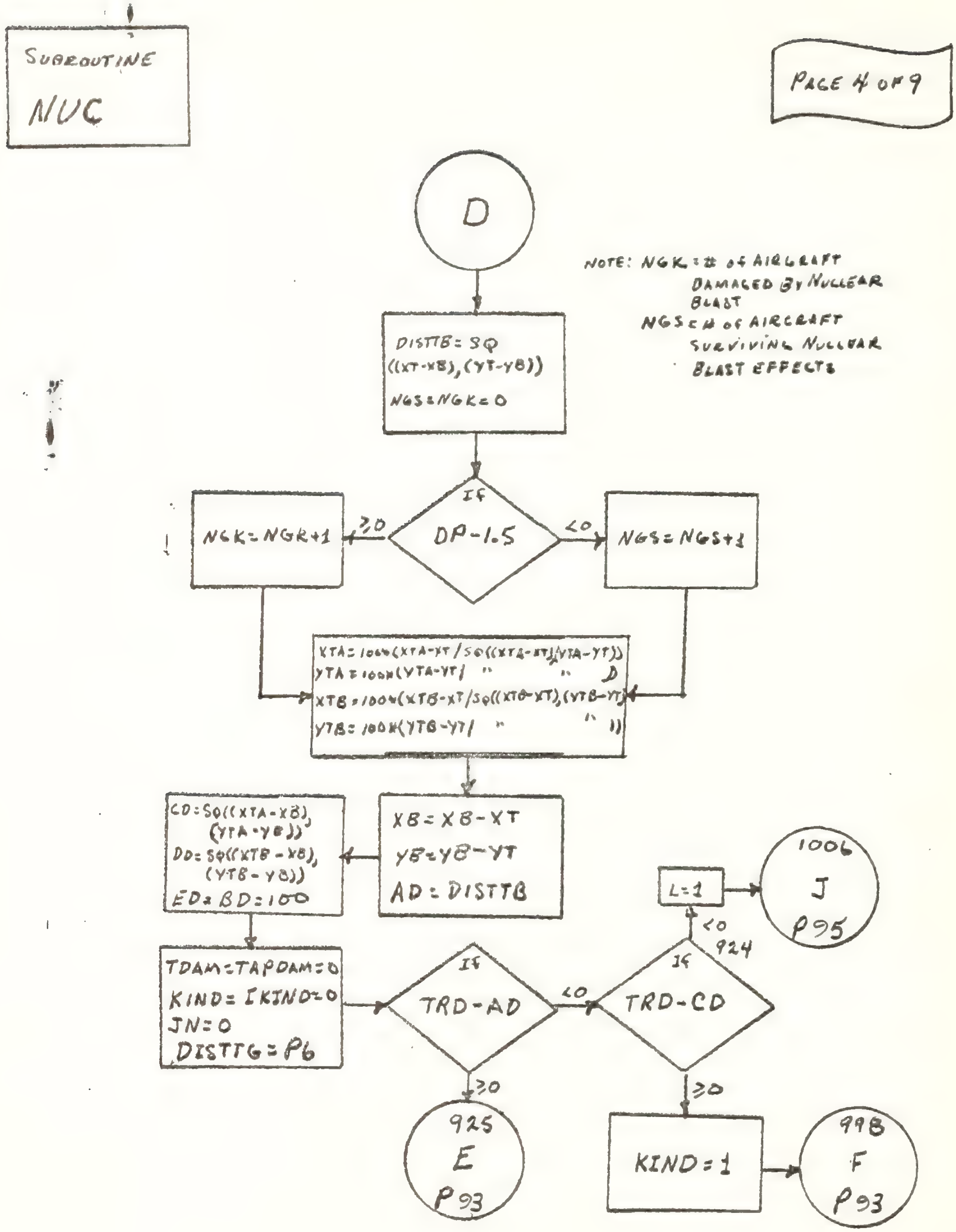

SUBROUTINE
ATTACK
PAGE 33 OF 33
P''
643
IGFLG
=1,2
=3,4
3333
IBUG=0
638
IF
P4-2
≥0
<0
640
IF
IGF-2
>0
<0
=0
Q''
644
IF
RANF(-1.) -
SPG(P3)
>0
≤0
333
PRINT
ATTACK
VARIABLES
IF
IBUG-3
=0
≠0
909
K=JJ+1
646
IGFLG =
IGFLG+2
IBUG=3
100
CONTINUE
554
Y'''
P62
625
S''
P57
≤0
645
IF
IGD-1
>0
7800
CALL PRINT1
(0,0,0,0,0,0,3)
JKT=1
U''
R''
RETURN
999
PRINT
INTERACTION
VARIABLES
T''
991
CALL PLOT1
(XA, YA, XG, YG,
XT, YT, IMAX
IGMAX, ITMAX)
V''



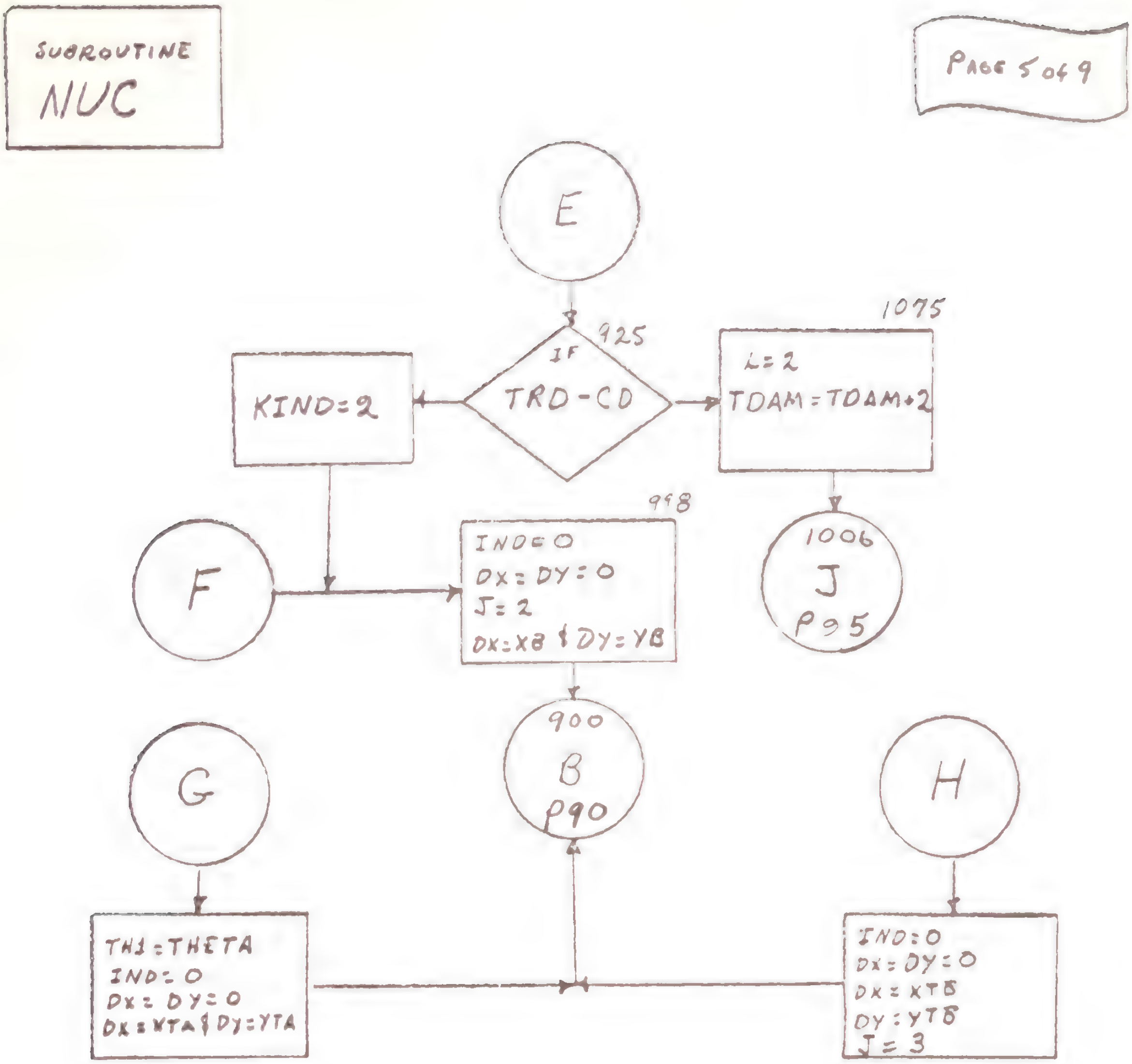
SUBROUTINE NUC
PAGE 1 OF 9
BEGIN
SQ(A,B)=SQRTF(A*A+B*B)
SQB(F,G,H)=SQRTF(F*F+G*G+H*H)
ASF(B,C,D)=ASINF(B*C/D)
JNUC
=1
=2
961
IF PK-RANF(-1.)
<0
≥0
1080 K 95
TDAM=TDAM+1
DISTTG=PG
J=IND=0 DX=DY=0
DX=XT-XG
DY=YT-YG
IF ITMØV
>0
900 B P90
917
XSPD=0
CEP=DISTTG/200.+XSPD/(1.6*DISTTG/100)
SIGMA=CEP*.847
PHIZ=359.999*RANF(-1.)
SUM=0.0
A
919
I=1,12
SUM=SUM+RANF(-1.)
D P92
XS=SUM-6.0
RØ=ABSF(XS*SIGMA)
920
XB=XT+RØ*COSF(PHIZ/57.295)
YB=YT+RØ*SINF(PHIZ/57.295)
DISTGB=SQB((XG-XB),(YG-YB),(ZG-ZB))
RIKT=DISTGB/(1.6W)^(1/3)*3.28003
RF=LOGF(1000./RIKT)
DP=EXPF(A+B*RF+C*RF*RF)

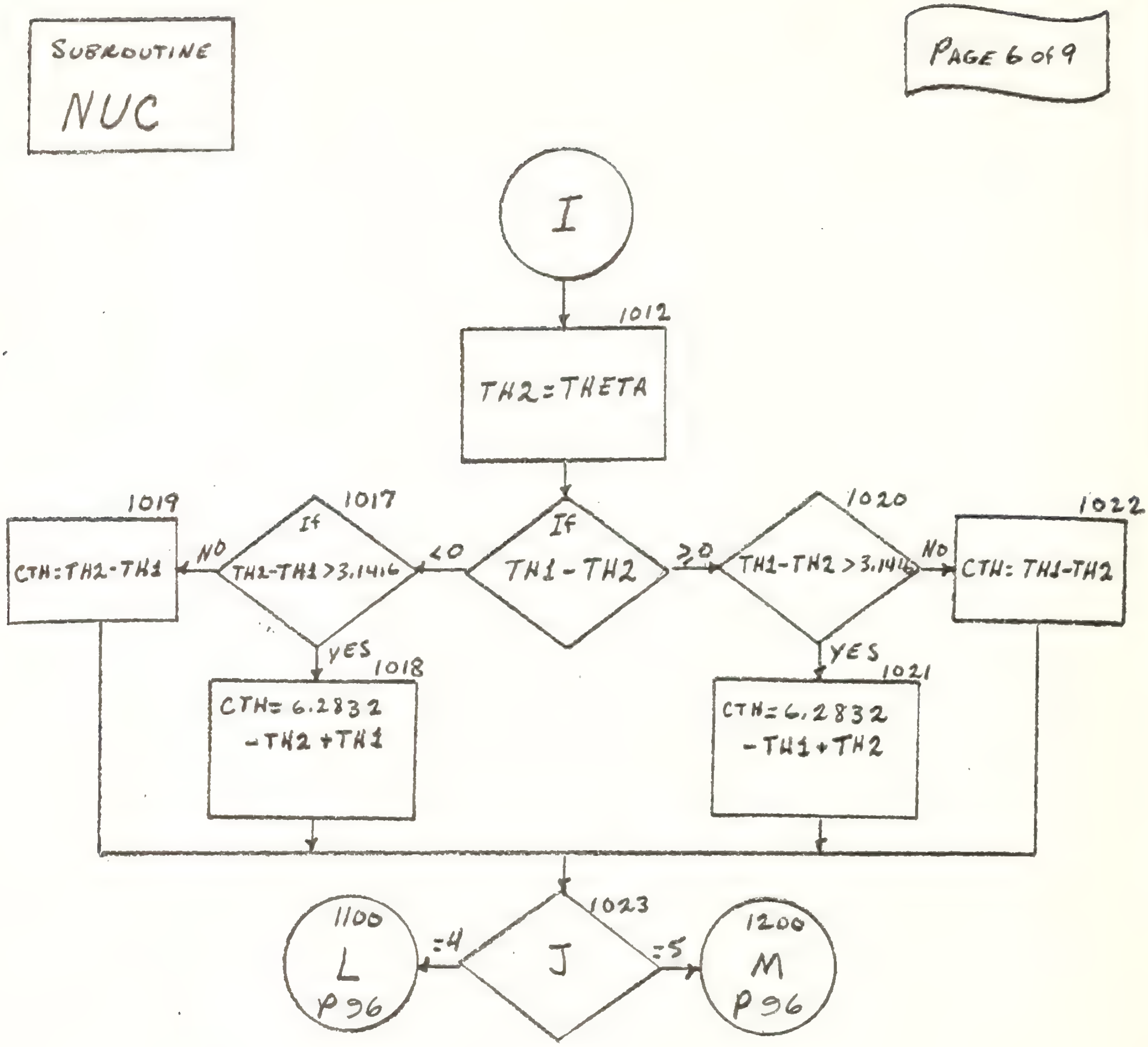

SUBROUTINE
NUC
PAGE 2 of 9
B
900
J=J+1
901
IF
DX
<0
DX=ABSF
(DX)
IND=IND
+2
≥0
902
IND=
IND+1
904
903
IF
DY
<0
DY=ABSF
(DY)
IND=IND
+2
≥0
THETA =
ATNF(DY
/DX)
IND
=1
907
C
P91
THETA=
3.1416 +
THETA
=3
=2
909
THETA =
6.2832-
THETA
908
THETA=
3.1416-
THETA


SUBROUTINE
NUC
PAGE 3 of 9
C
907
J
=3
=1
=2
=4,5
1011
G
P93
1012
I
P94
914
THETAT = THETA
THETAG = THETA
IND=0 $ DX=DY=0
DX=XTA-XT
DY=YTA-YT
900
B
P90
IF
THETAG - THETAT
≥0
<0
ETA =
THETAT-THETAG
915
913
RELSPD= SQRTF
(TSPD*TSPD+GSGM
*GSGM-2.0*TSPD
*GSGM*COSF(ETA)
ETA= 6.2832-
THETAT+THETAG
916
RLTHETA=SINF
(TSPD/RELSPD*
SINF(ETA))
XSPD= ABSF
(RLTHETA*RELSPD)
918
A
P89



SUBROUTINE NUC
PAGE 4 OF 9
D
NOTE: NGK = # of AIRCRAFT DAMAGED BY NUCLEAR BLAST
NGS = # of AIRCRAFT SURVIVING NUCLEAR BLAST EFFECTS
DISTTB = SQ ((XT-XB), (YT-YB))
NGS = NGK = 0
IF
DP-1.5
≥0
NGK = NGR+1
<0
NGS = NGS+1
XTA = 100*(XTA-XT / SQ((XTA-XT), (YTA-YT))
YTA = 100*(YTA-YT / " ")
XTB = 100*(XTB-XT / SQ((XTB-XT), (YTB-YT))
YTB = 100*(YTB-YT / " "))
XB = XB-XT
YB = YB-YT
AD = DISTTB
CD = SQ((XTA-XB), (YTA-YB))
DD = SQ((XTB-XB), (YTB-YB))
ED = BD = 100
TDAM = TAPDAM = 0
KIND = IKIND = 0
JN = 0
DISTTG = P6
IF
TRD-AD
<0
IF
TRD-CD
924
<0
L = 1
1006
J
P95
≥0
925
E
P93
≥0
KIND = 1
998
F
P93

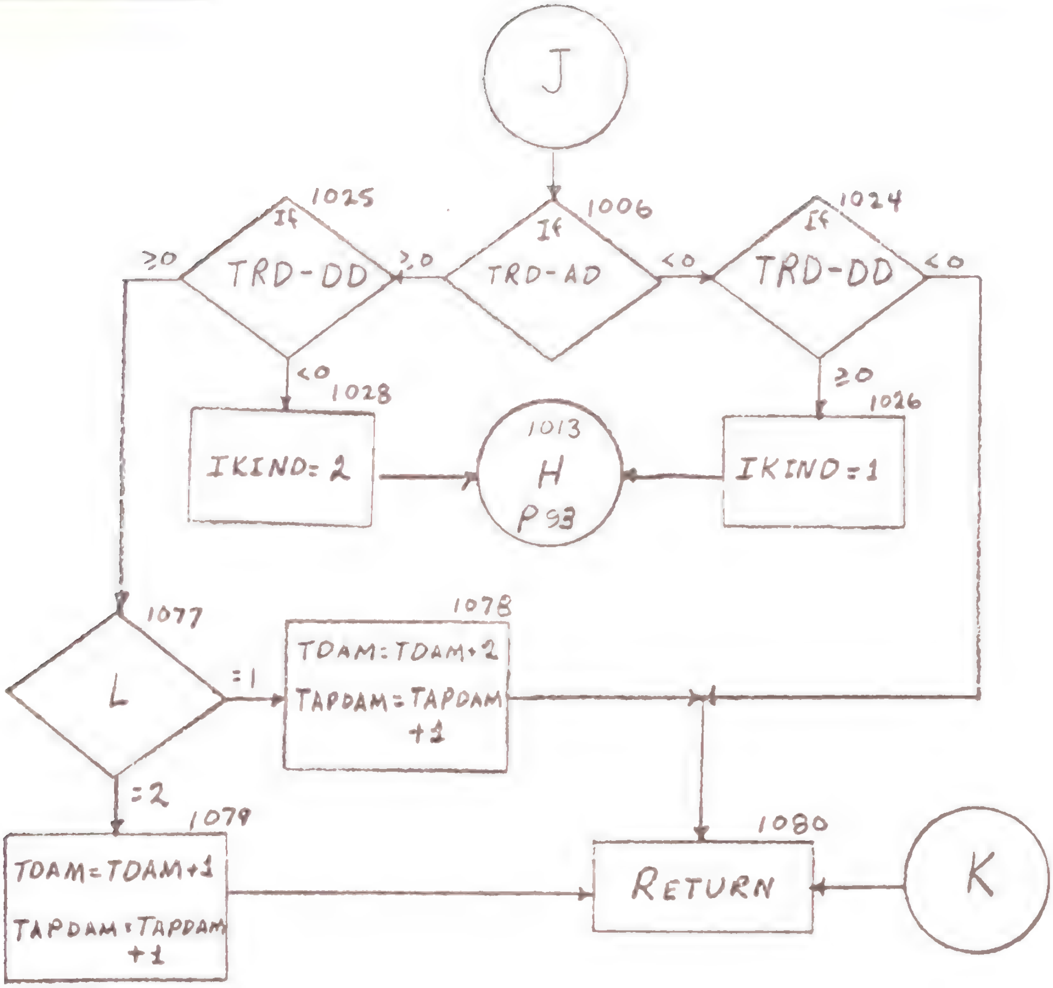

SUBROUTINE
NUC
PAGE 5 of 9
E
925
IF
TRD-CD
KIND=2
1075
L=2
TDAM=TDAM+2
998
IND=0
DX=DY=0
J=2
DX=XB $ DY=YB
F
1006
J
P95
900
B
P90
G
H
TH1=THETA
IND=0
DX=DY=0
DX=XTA $ DY=YTA
IND=0
DX=DY=0
DX=XTB
DY=YTB
J=3

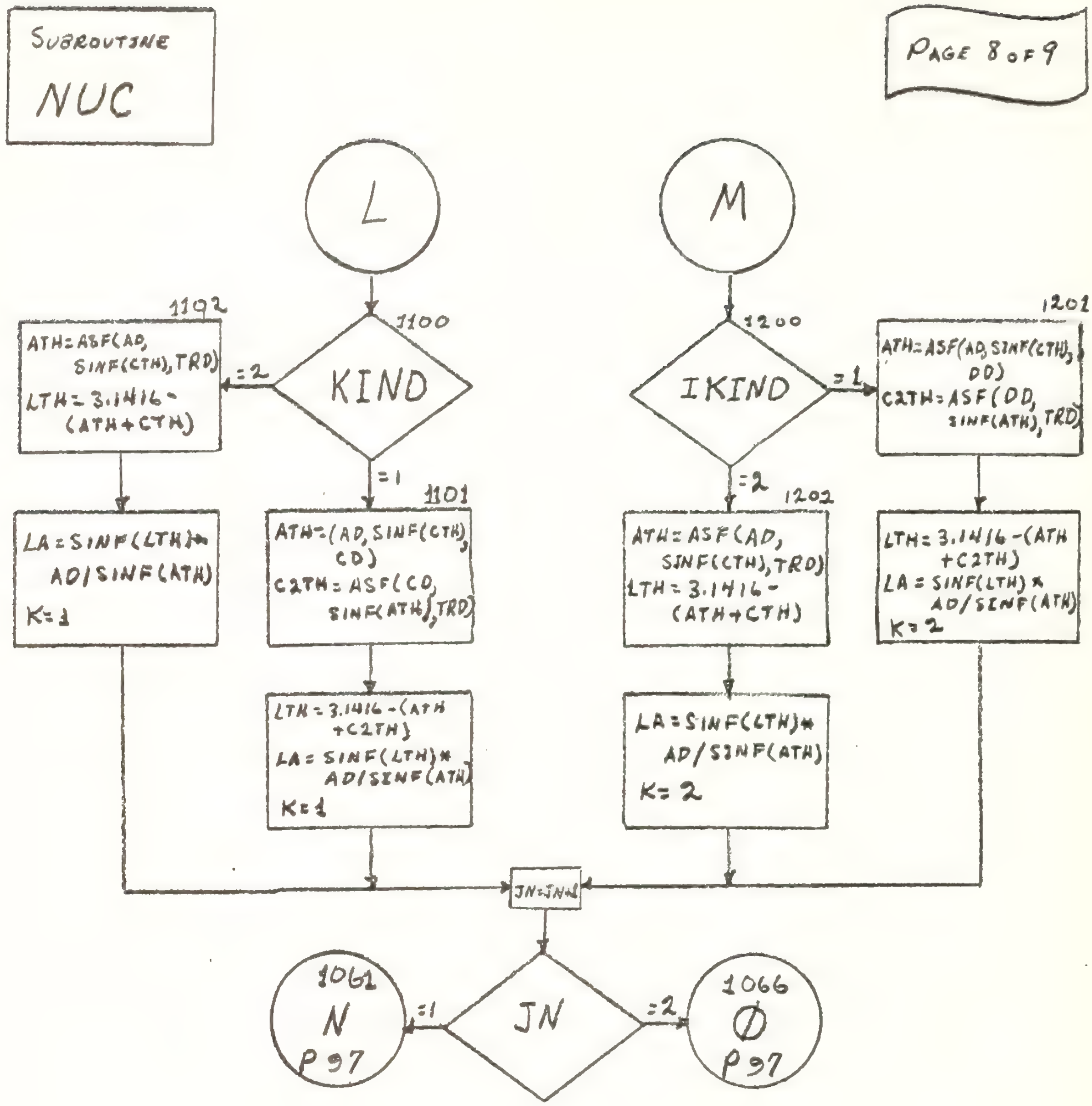


SUBROUTINE
NUC
PAGE 6 of 9
I
1012
TH2=THETA
If
TH1 - TH2
<0
≥0
1017
If
TH2-TH1 > 3.1416
NO
YES
1019
CTH:TH2-TH1
1018
CTH= 6.2832
-TH2 +TH1
1020
TH1-TH2 > 3.1416
NO
YES
1022
CTH: TH1-TH2
1021
CTH= 6.2832
-TH1+TH2
1023
J
=4
=5
1100
L
P96
1200
M
P96

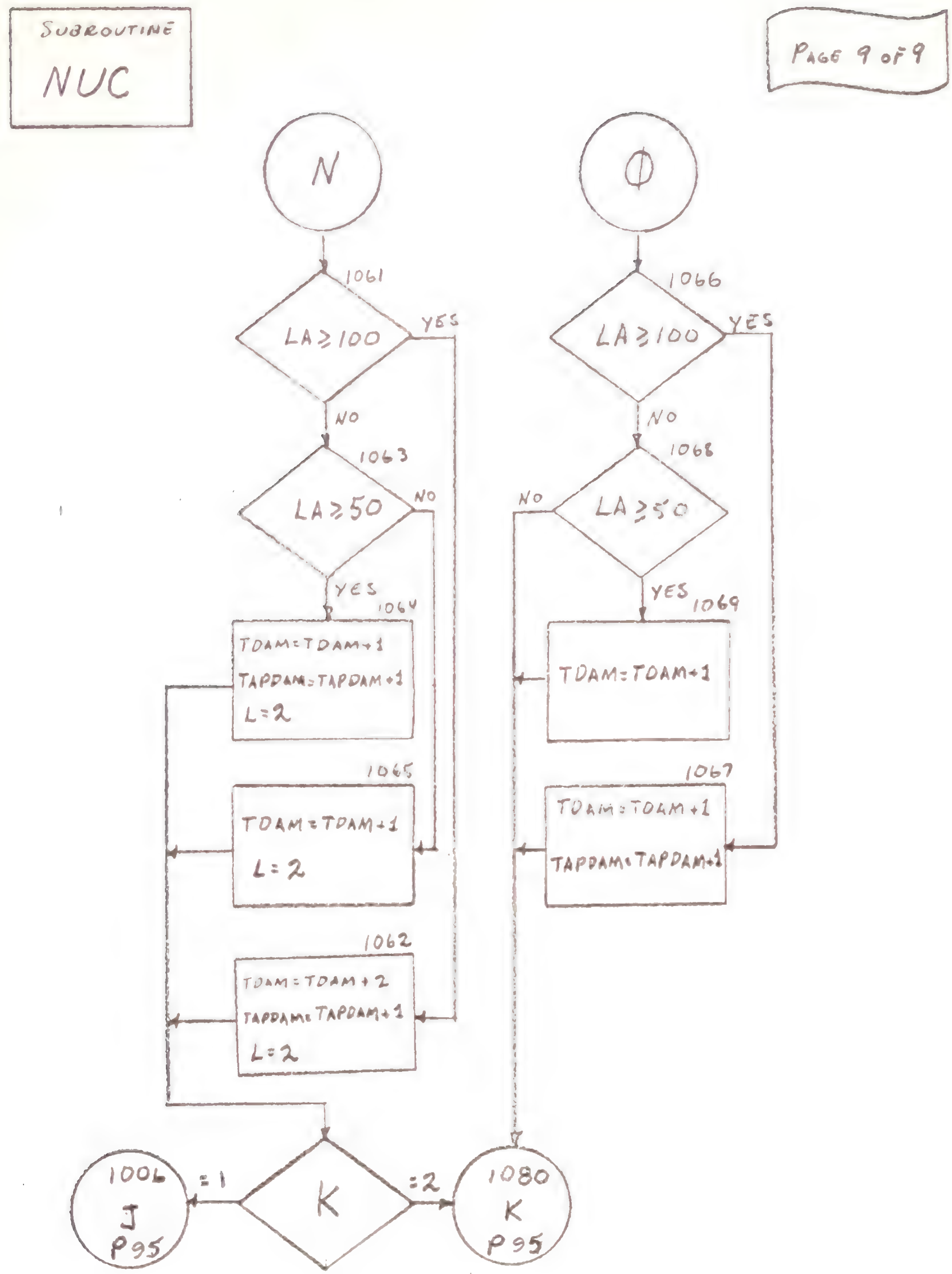
SUBROUTINE

NUC





SUBROUTINE
NUC
PAGE 8 OF 9
L
M
1100
KIND
=2
1102
ATH=ASF(AD, SINF(CTH), TRD)
LTH=3.1416-(ATH+CTH)
LA=SINF(LTH)*AD/SINF(ATH)
K=1
=1
1101
ATH=(AD, SINF(CTH), CD)
C2TH=ASF(CD, SINF(ATH), TRD)
LTH=3.1416-(ATH+C2TH)
LA=SINF(LTH)*AD/SINF(ATH)
K=1
1200
IKIND
=1
1201
ATH=ASF(AD, SINF(CTH), DD)
C2TH=ASF(DD, SINF(ATH), TRD)
LTH=3.1416-(ATH+C2TH)
LA=SINF(LTH)*AD/SINF(ATH)
K=2
=2
1202
ATH=ASF(AD, SINF(CTH), TRD)
LTH=3.1416-(ATH+CTH)
LA=SINF(LTH)*AD/SINF(ATH)
K=2
JN=JN+1
JN
=1
1061
N
P 97
=2
1066
Ø
P 97


SUBROUTINE
NUC
PAGE 9 OF 9
N
Φ
1061
LA ≥ 100
YES
NO
1063
LA ≥ 50
NO
YES
1064
TDAM=TDAM+1
TAPDAM=TAPDAM+1
L=2
1065
TDAM=TDAM+1
L=2
1062
TDAM=TDAM+2
TAPDAM=TAPDAM+1
L=2
1066
LA ≥ 100
YES
NO
1068
LA ≥ 50
NO
YES
1069
TDAM=TDAM+1
1067
TDAM=TDAM+1
TAPDAM=TAPDAM+1
K
=1
=2
1006
J
P95
1080
K
P95

APPENDIX II

This appendix contains the FORTRAN listing of the ARMREC computer program. All major divisions of the program are identified and comments are inserted into the listing where appropriate.

```
-COOP,,VANLJONS,0/49/S/56/57/7/E/45=54,60 ,100000,,PROGRAM ARMREC.
-FTN,E.
      PROGRAM ARMREC
       TYPE INTEGER RNNUM,TA,FT,RLT,P1,P2,P3,P4
      COMMON/RANGE1/N,X,Y,JKT
     2      /ELEV1/PHI(12,25,25),MMAX(12),NMAX(12),BCX1,BCX2,BCX3,
     3          BCX4,BCY1,BCY2,BCY3,XL,YL,ZE,BCX,BCY,TPD
     5      /ATT1/XTI,YTI,ZTI,XAI,YAI,ZAI,XGI,YGI,ZGI,TT,TS,TTA,[illegible],
     6IGFLG,RNNUM,GSM,TSPD,P2,P1,GSGM,P4,P3,[illegible],XA,YA,ZH,XG,YG,ZGH,
     7IMAX,IGMAX,ITMAX,JTYP,TA,FT,RLT,RMAX,RMIN,P,IKILL,[illegible],TFT,SPT,
     7SPA,SPG,TSV1,TF,TX,GSMX,NS,NG,ZA,ZG,HTR,TOF
     C,P5,P6
     D,XT,YT,ZT
     D,INVC,INV,RC,I9
     9      /ALOS/ ITSE,ICAT,DIS,ATR,ISD,IGD
     9      /CONT1/AC,AD,IRCAL,IER
     9      /NUC1/A,B,C,W,PK,TRD,TDAM,TAPDAM
     B      /BCK1/XMAX,XMIN,YMAX,YMIN,ITD
      DIMENSION XT(300),YT(300),ZT(300),XA(300),YA(300),ZA(300),ZH(300),
     1          XG(300),YG(300),ZG(300),ZGH(300),WPNA(4),WPNB(4),JTYP(4)
     2         ,TA(4),FT(4),RLT(4),RMAX(4),RMIN(4),AFIELD(16),SPA(25),SP
     3          T(25),SPG(25),P(4,25),TFT(4),TF(4),TX(4)
     4         ,INVC(2),INV(2)
      ISD=IGD=IPP=0
C
C     KKX IS USED TO INITIALIZE A UNIFORMLY DISTRIBUTED RANDOM NUMBER
C     GENERATOR(RANF). IT MUST BE SET FOR EACH REPLICATION TO AVOID
C     DUPLICATION.
C
      KKX=12
      DO 2346 I=1,KKX
 2346 GRABMF=RANF(-1.)
      READ 2000,ITL
C
C     ITL= TIME LENGTH OF GAME IN MINUTES ( LESS THAN 99)
C
```

```
      READ 1036,INVC,INV
C
C     INVC AND INV ARE,RESPECTIVELY, THE NUMBER OF NUCLEAR AND CONVENTIO
C     MISSLES ABOARD EACH ATTACK HELICOPTER.
C
      READ 2002,SPA,SPT,SPG
C
C     SPA,SPT,SPG ARE MATRICES OF SIGHTING PROBABILITIES FOR THE SCOUT.
C     TARGET, AND ATTACK ELEMENTS. PROBABILITIES ARE FOR 100 METER
C     INCREMENTS TOO 2500 METERS.
C
      READ 1004,DELV,ATR,HTR,TOF
C
C     DELV=MAXIMUM ALTITUDE CHANGE DURING A POP UP(FEET).
C      ATR=DESIRED ATTACK RANGE(METERS).
C      HTR=HELO AQUISITION TIME(SEC.).
C      TOF=MISSLE SPEED(METERS PER SEC.).
C
      READ 1000,NRUN
C
C     NRUN   NUMBER OF RUNS
C
      READ 1000,KMAX,XMIN,YMIN,XMAX,YMAX,TI,TPD
C
C     KMAX=NUMBER OF POLYNOMIALS TO APPROXIMATE THE TERRAIN.
C     XMIN=X COORDINATE OF LEFT EDGE OF THE TERRAIN.
C     YMIN=Y COORDINATE OF BOTTOM EDGE
C     XMAX=X COORDINATE OF RIGHT EDGE
C     YMAX=Y COORDINATE OF TOP EDGE
C     TI=GAME TIME INTERVAL (SECONDS)
C     TPD=DIMENSION OF TERRAIN SQUARE APPROXIMATED BY ONE POLYNOMIAL.
C
      IM= 60 *ITL / TI
      IF(KMAX.GT.12)1,2
    1 PRINT 1002
      STOP
```

```
    2 DO 123 I=1,KMAX
      READ 1014,MMAX(I),NMAX(I)
C
C     MMAX(I) = DEGREE OF X FOR THE ITH POLYNOMIAL.
C     NMAX(I)= DEGREE OF Y FOR THE ITH POLYNOMIAL
C
      MMAXX=MMAX(I)+1 $ NMAXX=NMAX(I)+1
      MMAX(I)=MMAX(I)+1 $ NMAX(I)=NMAX(I)+1
      IF(MMAX(I).GT.25.OR.NMAX(I).GT.25)122,123
  122 PRINT 1016
  123 READ 1018,((PHI(I,K,L),L=1,NMAXX),K=1,MMAXX)
C
C     PHI(I,K,L) IS THE COEFFICIENT FOR THE KTH DEGREE OF X THE LTH .
C     DEGREE OF Y, FOR THE ITH POLYNOMIAL.
C
      BCX=XMIN-TPD $ BCY=YMIN-TPD
      BCX1=XMIN+TPD/2. $ BCX2=BCX1+TPD $ BCX3=BCX2+TPD $ BCX4=BCX3+TPD
      BCY1=YMIN+TPD/2. $ BCY2=BCY1+TPD $ BCY3=BCY2+TPD
      READ 1018,A,B,C,W
      READ 1004,PK,TRD
      DO 691 M=1,NRUN
      RNNUM=M
      READ 1000,NWPTYP
C
C     NWPTYP NUMBER OF WEAPON TYPES
C
      DO 3 I=1,NWPTYP
      READ 1010,WPNA(I),WPNB(I),JTYP(I),TA(I),FT(I),RLT(I),RMAX(I),
     1          RMIN(I),TFT(I)
C
C     WPNA,WPNB=TYPE OF WEAPON (COL.1-16)
C     JTYP   NUMBER OF WEAPONS OF TYPE I
C     TA=TIME REQUIRED TO ACQUIRE TARGET (SEC.)
C     FT=FIRING TIME BEFORE RELOADING (SEC.)
C     RLT=TIME REQUIRED TO RELOAD (SEC.)
C     RMAX=MAXIMUM RANGE OF WEAPON (METERS)
```

```
C     RMIN=MINIMUM RANGE OF WEAPON (METERS)
C
    3 READ 1012,(P(I,K),K=1,25)
C
C     P(I,K) IS THE KILL PROBABILITY FOR WEAPON TYPE I AT THE KTH 100
C     METER INTERVAL.
C
      PRINT 1019
      PRINT 1020,(WPNA(I),WPNB(I),I=1,NWPTYP)
      PRINT 1022,(JTYP(I),I=1,NWPTYP)
      PRINT 1024,(TA(I),I=1,NWPTYP)
      PRINT 1026,(FT(I),I=1,NWPTYP)
      PRINT 1028,(RLT(I),I=1,NWPTYP)
      PRINT 1030,(RMIN(I),I=1,NWPTYP)
      PRINT 1032,(RMAX(I),I=1,NWPTYP)
      READ 1034,(AFIELD(I),I=1,16)
C
C     THIS READS IN THE TITLE OF THE RUN
C
      IRCAL=1
      READ 1004,GMAX,RC,RD,GS,TSPD,GSG,TSPDM,GSX
C
C     GMAX=MAXIMUM POSITIVE G TO BE USED BY THE AIRCRAFT.
C     RC=MAXIMUM RATE OF CLIMB (FT./MIN)
C     RD=MAXIMUM RATE OF DESCENT (FT./MIN)
C     GS=SCOUT SECTION GROUND SPEED(KNOTS)
C     GSG=ATTACK SECTION GROUND SPEED(KNOTS)
C     GSX=MAXIMUM SCOUT SECTION GROUND SPEED(KNOTS)
C     TSPDM = EMERGENCY TARGET SPEED
C
      TSPDM=TSPDM*.44704
      TSPD=TSPD*.44704
      GSGM= GSG*0.514444
      GSM=GS*0.514444
      GSMX=GSX*0.514444
      I9=DELV/RC*60
```

```
      READ 1036,IMARK,ISEP,IPOP,IKILL
      PRINT 1036,IMARK,ISEP,IPOP,IKILL
C
C
C                                FLIGHT PATH
C                            SCOUT      ATTACK GROUP
C                  IMARK = 1   PP         PP
C                        = 2   NOE        NOE
C                        = 3   PP         NOE
C                        = 4   NOE        PP
C
C                  PP=PREPLANNED FLIGHT PATH
C                  NOE=NAP OF THE EARTH FLIGHT PATH
C
C     IPOP= MIN ELAPSED TIME BETWEEN POPUPS
C     ISEP IS A PARAMETER USED TO EFFECT TIME AND DISTANCE SEPERATION
C     BETWEEN ATTACK AND SCOUT AIRCRAFT.
C
C                  IKILL =1 SCOUT SECTION IS THE PRIDRITY TARGET
C                        =2 ATTACK SECTION IS THE PRIORITY TARGET
C                        =3 LOWEST SURVIVAL PROBABILITY DETERMINES
C                              TARGET PRIORITY
C                        =4 ALL AIRCRAFT WILL BE TAKEN UNDER FIRE
C                              SIMULTANEOUSLY, IF POSSIBLE.
C
C
    7 PRINT 1040,(AFIFLD(I),I=1,4)
      PRINT 1048,(AFIFLD(I),I=5,16)
      GO TO(109,108,109,108),IMARK
   82 PRINT 1046 $ STOP
  109 I=1
  133 READ 1042,XA(I),YA(I),ZH(I),ISTP
      IF(ISTP.EQ.1)132,131
  131 ZH(I)=ZH(I)*.304801
C
C     XA(I)=PREPLANNED X COORDINATE IN THE ITH TIME INTERVAL
```

```
C     YA(I)=PREPLANNED Y COORDINATE
C     ZH(I)=PREPLANNED Z COORDINATE
C
      XL=XA(I) $ YL=YA(I) $ CALL ELEV
      IF(ZH(I)-ZE)82,82,81
   81 I=I+1 $ GO TO 133
  132 IMAX=I-1 $ GO TO 117
  108 I=1
  134 READ 1042,XA(I),YA(I),ZH(I),ISTP
      ZH(I)=ZH(I)*.304801
      IF(ISTP.EQ.1)136,135
  135 XL=XA(I) $ YL=YA(I) $ CALL ELEV
      ZA(I)=ZE
      I=I+1 $ GO TO 134
  136 IMAX=I-1
      AC=RC*.00508001/GSM $ AD=-RD*.00508001/GSM
      CALL CONTOUR(IMAX,XA,YA,ZA,ZH,2)
C
C     COMPUTES NOE FLIGHTPATH ALTITUDES.
C
  117 GO TO (100,97,97,100),IMARK
   97 I=1
  137 READ 1042,XG(I),YG(I),ZGH(I),ISTP
      ZGH(I)=ZGH(I)*.304801
      IF(ISTP.EQ.1)139,138
  138 XL=XG(I) $ YL=YG(I) $ CALL ELEV
      ZG(I)=ZE
      I=I+1 $ GO TO 137
  139 IGMAX=I-1
  107 AC=RC*.00508001/GSGM $ AD=-RD*.00508001/GSGM
      CALL CONTOUR(IGMAX,XG,YG,ZG,ZGH,2) $ GO TO 91
  100 I=1
  140 READ 1042,XG(I),YG(I),ZGH(I),ISTP
      IF(ISTP.EQ.1)142,141
  141 ZGH(I)=ZGH(I)*0.304801
      XL=XG(I) $ YL=YG(I) $ CALL ELEV
```

```
      IF(ZGH(I)-ZE)82,82,101
  101 I=I+1 $ GO TO 140
  142 IGMAX=I-1
   91 IP=1
  143 READ 1042,XT(IP),YT(IP),ZT(IP),ISTP
      IF(ISTP.EQ.1)145,144
C
C     XT(I)=PREPLANNED X COORDINATE IN THE ITH TIME INTERVAL
C     YT(I)=PREPLANNED Y COORDINATE
C     ZT(I)=PREPLANNED Z COORDINATE
C
  144 XL=XT(IP) $ YL= YT(IP) $ CALL ELEV
      ZT(IP)=ZE
      IP=IP+1 $ GO TO 143
  145 ITMAX=IP-1
      CALL PLOT1 (XA,YA,XG,YG,XT,YT,IMAX,IGMAX,ITMAX)
      JKT=2 $ IAT=1
   96 DTG=-ISEP/GSGM
      TG=DTG
      TT=TTA=T1=T2=0
   18 DO 22 IJ=1,IM
  800 TTA=TTA+TI
      CALL RANGE(XA,YA,ZH,IMAX,TTA,GSM)  $ NS=N+1
      GO TO (69,62),JKT
   62 XAI=X $ YAI=Y
      ZAI=ZH(N)+(XAI-XA(N))*(ZH(N+1)-ZH(N))/(XA(N+1)-XA(N))
      TT=TT+TI
      CALL RANGE(XT,YT,ZT,ITMAX,TT,TSPD)
      GO TO (69,63),JKT
   63 XTI=X $ YTI=Y
      XL=XTI $ YL=YTI $ CALL ELEV
      ZTI=ZE
      CALL ALOS(XAI,YAI,ZAI,XTI,YTI,ZTI)
      P2=ITSE $ P1=ICAT $ P5=DIS
C
C     ITSE=1   LOS NONEXISTENT
```

```
C         =2   LOS EXISTS
C
      TG=TG+TI
      IF(TG.GE.0)171,172
  171 CALL RANGE(XG,YG,ZGH,IGMAX,TG,GSGM) $ NG=N+1
      GO TO (69,64),JKT
   64 XGI=X $ YGI=Y
      ZGI=ZGH(N)+(XGI-XG(N))*(ZGH(N+1)-ZGH(N))/(XG(N+1)-XG(N))
      CALL ALOS(XGI,YGI,ZGI,XTI,YTI,ZTI)
      P4=ITSE $ P3=ICAT $ P6=DIS
      IF(P3.GT.25)172,163
  172 IGFLG=2 $ GO TO 176
  163 IF(P4.EQ.2)164,161
  161 IGFLG=2
  173 IF(P1.GT.25)174,162
  174 ISFLG=2 $ IF(IGFLG.EQ.2)18,21
  176 IF(P1.GT.25)18,162
  164 IF(RANF(-1.).LE.SPG(P3))165,166
  165 IGFLG=2 $ GO TO 167
  166 IGFLG=0
  167 IF(RANF(-1.).LE.SPT(P3))168,169
  168 IGFLG=IGFLG+1 $ GO TO 173
  169 IGFLG=IGFLG+2 $ GO TO 173


C
C     THE SAME CONVENTION APPLIES TO THE CODING OF IGFLG AS IS USED
C     FOR ISFLG.
C
  162 IF(P2.EQ.2)10,16
   10 IF(RANF(-1.).LE.SPA(P1))11,12
   11 ISFLG=2  $  GO TO 13
   12 ISFLG=0
   13 IF(RANF(-1.).LE.SPT(P1))14,15
   14 ISFLG =ISFLG+1 $ GO TO 160
   15 ISFLG=ISFLG+2
```

```
C
C          DOES    A/C     /    TARGET      SEE THE OTHER
C     ISFLAG=1       N              Y
C            2       N              N
C            3       Y              Y
C            4       Y              N
C
  160 GO TO (300,300,21,21),ISFLG
   16 ISFLG=2 $ GO TO 300
C
C     POP UP CONTROL FOR SCOUT AIRCRAFT IS EXERCISED BELOW.
C
  300 IF(TTA-T1-IPOP)201,201,19
  201 IF(ISFLG.EQ.2.AND.IGFLG.EQ.2)22,21
   19 IF(IPP)190,190,191
  191 IF(IPP-I9)355,355,192
  190 PRINT 4040
      IPP=IPP+1 $ ZAI=ZAI+RC*.00508*IPP $ ZAII=ZAI $ IF(P2-1)23,20,23
  355 PRINT 4040
      IPP=IPP+1 $ ZAI=ZAII+RC*.00508*IPP $ ZAII=ZAI $ IF(P2-1)23,20,23
  192 PRINT 4040
      ZAI=ZAII+RC*.00508*IPP $ IPP=0 $ T1=TTA $ IF(P2-1)23,20,23
   20 CALL ALOS(XAI,YAI,ZAI,XTI,YTI,ZTI)
      P2=ITSE $ P1=ICAT
      IF(P2-2)177,121,121
  177 IF(IGFLG.EQ.2)22,21
   23 IF(RANF(-1.).LE.SPA(P1))240,27
   27 IF(ISFLG.EQ.2.AND.IGFLG.EQ.2)22,21
  121 T1=TTA
      IF(RANF(-1.)-SPA(P1))194,194,195
  194 ISFLG=2 $ GO TO 196
  195 ISFLG=0
  196 IF(RANF(-1.)-SPT(P1))197,197,198
  197 ISFLG=ISFLG+1 $IPP=0 $  GO TO 199
  198 ISFLG=ISFLG+2 $ IPP=0
  199 GO TO (21,18,21,21),ISFLG
```

```
240 ISFLG=ISFLG+2 $ T1=TTA    $ IPP=0
 21 TSV1=TTA    $ IPP=0
    PRINT 3738,XGI,YGI,ZGI,TG,GSGM,P4,P3
    PRINT 3737,XAI,YAI,ZAI,TTA,GSM,XTI,YTI,ZTI,TT,TSPD,P2,P1
    PRINT 3739,ISFLG,IGFLG
3739 FORMAT(18X,8HISFLG = ,I10,3X,8HIGFLG = ,I10//)
    CALL ATTACK
    GO TO(69,22),JKT
 22 CONTINUE
 69 CALL PLOT1(XA,YA,XG,YG,XT,YT,IMAX,IGMAX,ITMAX)
691 CONTINUE
1000 FORMAT(I10,6F10.0)
1002 FORMAT(1H1////77H NUMBER OF APPROXIMATING TERRAIN POLYNOMIALS EXCE
   1EDS 12. THIS RUN TERMINATED.)
1004 FORMAT(8F10.0)
1010 FORMAT(2A8,4X,4I5,3F10.0)
1012 FORMAT(10F8.4)
1014 FORMAT(2I10)
1016 FORMAT(1H1////78H MAXIMUM DEGREE OF THE TERRAIN POLYNOMIAL IS EXCE
   1EDED. THIS RUN IS TERMINATED.)
1018 FORMAT(4E20.10)
1019 FORMAT(1H1///12X,30HSUMMARY OF TARGET WEAPON DATA. //)
1020 FORMAT(1H0,11HWEAPON TYPE,10X,12A8)
1022 FORMAT(1X,17HNUMBER OF WEAPONS,I10,5I16)
1024 FORMAT(1X,22HACQUISITION TIME (SEC),I5,5I16)
1026 FORMAT(1X,17HFIRING TIME (SEC),I10,5I16)
1028 FORMAT(1X,20HRELOADING TIME (SEC),I7,5I16)
1030 FORMAT(1X,22HMINIMUM RANGE (METERS),F6.0,5F16.0)
1032 FORMAT(1X,22HMAXIMUM RANGE (METERS),F6.0,5F16.0)
1034 FORMAT(4A8)
1036 FORMAT(7I10)
1040 FORMAT(////31X,10HSUMMARY OF/28X,4A8)
1042 FORMAT(3F20.10,I5)
1046 FORMAT(1H1,74HTERRAIN ELEVATION EXCEEDS PREPLANNED AIRCRAFT HEIGHT
   1. THIS RUN TERMINATED.)
1048 FORMAT(19X,10HMISSION -.,4A8/(29X,4A8))
```

```
2000 FORMAT(I2)
2002 FORMAT(5F10.8)
2006 FORMAT(40X,F20.10)
3738 FORMAT(10X,6HXGI = ,F10.3,5X,6HYGI = ,F10.3,5X,6HZGI = ,F10.3//10X
    26H TG = ,F10.3,4X,7HGSGM = ,F10.3//11X,5HP4 = ,I10,6X,5HP3 = ,I10
    3/)
3737 FORMAT(//
    A        10X,6HXAI = ,F10.3,5X,6HYAI = ,F10.3,5X,6HZAI = ,F10.3//10X
    1,6HTTA = ,F10.3,5X,6HGSM = ,F10.3//10X,6HXTI = ,F10.3,5X,6HYTI = ,
    2F10.3,5X,6HZTI = ,F10.3//10X,6H TT = ,F10.3,4X,7HTSPP = ,F10.3//11
    3X,5HP2 = ,I10,6X,5HP1 = ,I10/)
4040 FORMAT(//5X,18HA/C HAS POPPED UP.//)
     END


     SUBROUTINE BNDCK(X,Y,IS)
C
C    THIS SUBROUTINE CHECKS FOR AN OUT OF BOUNDS CONDITION.
C
     COMMON/BCK1/XMAX,XMIN,YMAX,YMIN,ITD
     IF(X-XMAX)2,2,3
   2 IF(XMIN-X)20,20,3
  20 IF(Y-YMAX)30,30,3
  30 IF(YMIN-Y)4,4,3
   3 GO TO(5,6,7),IS
   5 PRINT8,X,Y $ STOP
   6 PRINT 9,X,Y $ STOP
   7 PRINT 10,X,Y $ STOP
   4 RETURN
   8 FORMAT(10X,43HAIRCRAFT OUT OF BOUNDS THIS RUN TERMINATED.,5X,6HXAI
    1 = ,F10.3,5X,6HYAI = ,F10.3)
   9 FORMAT(10X,43HAIRCRAFT OUT OF BOUNDS THIS RUN TERMINATED.,5X,6HXGI
    1 = ,F10.3,5X,6HYGI = ,F10.3)
```

```
   10 FORMAT(10X,41HTARGFT OUT OF BOUNDS THIS RUN TERMINATED.,5X,6HXTI =
     1 ,F10.3,5X,6HYTI = ,F10.3)
      END



      SUBROUTINE RANGE(XB,YB,ZB,ITMAX,TA,GSM)
C
C     THIS SUBROUTINE DETERMINES ELEMENT POSITION.
C
      DIMENSION XB(300), YB(300), ZB(300)
      COMMON/RANGE1/N,XTI,YTI,JKT
      DELD=TA*GSM
   70 DIST=0.
      NPI=ITMAX-1
   10 DO 6 I=1,NPI
      DP=SQRTF((XB(I+1)-XB(I))**2+(YB(I+1)-YB(I))**2+(ZB(I+1)-ZB(I))**2)
      DIST=DIST+DP $ COS=(YB(I+1)-YB(I))/DP $ SIN=(XB(I+1)-XB(I))/DP
   50 IF(DIST-DELD)6,20,20
   20 IF(DIST.GT.DELD) 30,40
   40 XTI=XB(I+1) $ YTI=YB(I+1) $ N=I $ GO TO 90
   30 XTI=XB(I+1)-(DIST-DELD)*SIN
      YTI=YB(I+1)-(DIST-DELD)*COS                $ N=I $ GO TO 90
    6 CONTINUE
  160 PRINT 1000
 1000 FORMAT(15X,66HA GAME ELEMENT HAS TERMINATED ITS MOVEMENT. THIS RUN
     1 IS TERMINATED.)
      JKT=1
   90 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ALOS(XA,YA,ZA,XTI,YTI,ZTI)
C
C     THIS SUBROUTINE CHECKS FOR THE EXISTENCE OF AN UNINTERRUPTED LINE
C     OF SIGHT.
C
      COMMON/ELEV1/PT(7531),XL,YL,Z,PJ(3)
     1      /ALOS/ITSE,ICAT,DIS,ATR,ISD,IGD
      DIS=SQRTF((YA-YTI)**2+(XA-XTI)**2)
      PHI=(ZA-ZTI)/DIS
      RW=10.
      SIN=(XTI-XA)/DIS $ COS=(YTI-YA)/DIS
  110 ZL=PHI*RW+ZTI
      XL=XTI-RW*SIN $ YL=YTI-RW*COS      $ CALL ELEV
      IF(Z .GT. ZL) 140,100
  100 RW=RW+50
      IF(RW.GT.DIS) 150,110
  140 ITSE=1 $ ICAT=DIS/100 +1  $  RETURN
  150 ITSE=2 $ ICAT=DIS/100 +1  $  RETURN
      END


      SUBROUTINE ELEV
C
C     THIS SUBROUTINE DETERMINES THE TERRAIN ELEVATION, GIVEN X AND Y.
C
      COMMON/ELEV1/PHI(12,25,25),MMAX(12),NMAX(12),BCX1,BCX2,BCX3,BCX4,
     1            BCY1,BCY2,BCY3,XT,YT,ZT,BCX,BCY,TPD
C
C     DETERMINE WHICH POLYNOMIAL TO SOLVE.
C
      CALL BNDCK(XT,YT,3)
```

```
      INX=(XT-BCX)/TPD $ INY=(YT-BCY)/TPD
      I=INX+4*(INY-1)
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12)I
    1 X=XT-BCX1       $ Y=YT-BCY1      $ GO TO 30
    2 X=XT-BCX2       $ Y=YT-BCY1      $ GO TO 30
    3 X=XT-BCX3       $ Y=YT-BCY1      $ GO TO 30
    4 X=XT-BCX4       $ Y=YT-BCY1      $ GO TO 30
    5 X=XT-BCX1       $ Y=YT-BCY2      $ GO TO 30
    6 X=XT-BCX2       $ Y=YT-BCY2      $ GO TO 30
    7 X=XT-BCX3       $ Y=YT-BCY2      $ GO TO 30
    8 X=XT-BCX4       $ Y=YT-BCY2      $ GO TO 30
    9 X=XT-BCX1       $ Y=YT-BCY3      $ GO TO 30
   10 X=XT-BCX2       $ Y=YT-BCY3      $ GO TO 30
   11 X=XT-BCX3       $ Y=YT-BCY3      $ GO TO 30
   12 X=XT-BCX4       $ Y=YT-BCY3
C
C     SOLVE THE POLYNOMIAL, ZCOMP = F(X,Y)
C
   30 ZCOMP=0 $ MMAXX=MMAX(I) $ NMAXX=NMAX(I)
      DO 101 K=1,MMAXX
      IK=MMAX(I)+1-K
      IL=NMAXX $ IG=NMAXX-1
      POLY=PHI(I,IK,IL)
      DO 100 L=1,IG $ IL=NMAXX-L
  100 POLY=POLY*Y+PHI(I,IK,IL)
  101 ZCOMP=ZCOMP*X+POLY
      ZT=0.304801*ZCOMP
      RETURN
      END


      SUBROUTINE CONTOUR(IM,XT,YT,ZT,Z,NS)
C
```

```
C     THIS SUBROUTINE COMPUTES A NAP OF THE EARTH FLIGHT PLAN BETWEEN
C     POINTS
C
      DIMENSION XT(300),YT(300),ZT(300),Z(300)
      COMMON/CONT1/AC,AD,IRCAL,IER
      IMARK=1
      DO 7 I=NS,IM
      IF(IM.LE.I)10,9
    9 GO TO (10,11)IMARK
   10 IMARK=2
      GO TO 19
   11 IF(ZT(I).LE.ZT(I-1))19,17
   17 F=ZT(I)+3.
      IF(ZT(I).LE.ZT(I+1))20,18
   20 IMARK=2 $ GO TO 21
   18 IMARK=1 $ GO TO 21
   19 F=ZT(I)+10.
   21 S=(F-Z(I-1))/SQRTF((XT(I)-XT(I-1))**2+(YT(I)-YT(I-1))**2)
      IF(AC-S)22,15,15
   15 IF(S-AD)22,100,100
  100 Z(I)=F $ GO TO 7
   22 IF(S)43,44,44
   43 Z(I)=Z(I-1)+AD*SQRTF((XT(I)-XT(I-1))**2+(YT(I)-YT(I-1))**2)
      GO TO 7
   44 Z(I)=F
      J=0
   46 J=J+1
      Z(I-J)=F-AC*SQRTF((XT(I-J)-XT(I-J-1))**2+(YT(I-J)-YT(I-J-1))**2)
      IF(I-J-NS+1)48,48,47
   47 IF(((Z(I-J)-Z(I-J-1))/SQRTF((XT(I-J)-XT(I-J-1))**2+(YT(I-J)-
     *YT(I-J-1))**2)).GT.AC)45,7
   45 F=Z(I-J) $ GO TO 46
   48 GO TO (480,481),IRCAL
  480 PRINT 7010
 7010 FORMAT(1H1////77HAIRCRAFT CLIMB ANGLE INSUFFICIENT FOR TERRAIN CLE
     1ARANCE. THIS RUN TERMINATED.)
```

```
      STOP
  481 IER=1 $ RETURN
    7 CONTINUE $ IER=2 $ RETURN
      END


      SUBROUTINE TGSE(XTI,YTI,P6,XGI,YGI,XFI,YFI,ZFI,ZTI,IA)
C
C     THIS PROGRAM SELECTS A POSITION SUCH THAT FOR
C         IA = 1 AN LOS EXISTS TO A SPECIFIED POINT.
C         IA=2 AN LOS DOES NOT EXIST TO A SPECIFIED POINT.
C
      COMMON/ALOS/ITSE,ICAT,DIS,ATR,ISD,IGD
     A      /TGSE/XFI6,YFI6,ZFI6
     B      /ELEV1/PT(7531),XL,YL,ZE,PU(3)
      TYPE INTEGER XNS
      VECF(A,B,C,D)=(A-B)/SQRTF((A-B)*(A-B)+(C-D)*(C-D))
      AZXB=ATR $ XNS=1 $ XPN=XMS=0 $ ILM=1
   11 XFP=XTI+ATR*(XGI-XTI)/P6 $ XFI=XFP
      YFP=YTI+ATR*(YGI-YTI)/P6 $ YFI=YFP
      IF(IA-1)6,6,4
    6 XL=XFI $ YL=YFI $ CALL ELEV
      ZFI=ZE+50 $ IF(ILM-1)15,12,15
   15 CALL ALOS(XTI,YTI,ZTI,XFI,YFI,ZFI)
      IF(ITSE-2/IA)1,14,1
    1 ZFII=ZFI
   16 ZFII=ZFII+50
      CALL ALOS(XTI,YTI,ZTI,XFI,YFI,ZFII)
      IF(ITSE-2/IA)3,2,3
    2 ZFI=ZFII $ GO TO 14
    3 IF(ZFII-ZFI-200.)16,16,4
    4 GO TO (5,7),XNS
    5 XPN=XPN+1
```

```
      XFI=XFP+XPN*100*VECF(YGI,YTI,XGI,XTI)
      YFI=YFP-XPN*100*VECF(XGI,XTI,YGI,YTI)
      XPN=XPN-1   $ XNS=2
      GO TO 6
    7 XPN=XPN+1
      IF(XPN-ATR/300)8,8,9
    8 XFI=XFP-XPN*100*VECF(YGI,YTI,XGI,XTI)
      YFI=YFP+XPN*100*VECF(XGI,XTI,YGI,YTI)
      XNS=1  $ GO TO 6
    9 XMS=XMS+1  $ IF(XMS-8)10,10,13
   10 XPN=0  $ XNS=1
      ATR=ATR+50*(-1)**IA $ GO TO 11
   12 YFI6=YFI $ XFI6=XFI $ ZFI6=ZFI $ ILM=2 $ GO TO 15
   13 XFI=XFI6 $ YFI=YFI6 $ ZFI=ZFI6
   14 ATR=AZXB
      RETURN
      END



      SUBROUTINE RTE(NG,XG,YG,ZG,ZGH,XGI,YGI,ZGI,XFI,YFI,ZFI,GSGM)
C
C     SUBROUTINE DETERMINES X AND Y COORDINATES OF A PATH SUCH THAT
C     NAP OF THE EARTH FLIGHT IS POSSIBLE.
C
      COMMON/ELEV1/PT(7531),XL,YL,ZE,PU(3)
     1       /CONT1/AC,AD,IRCAL,IFR
     2       /RTE/IAA
      DIMENSION XG(300),YG(300),ZG(300),ZGH(300)
      DXBF(A,B,C,D)=SQRTF((A-B)*(A-B)+(C-D)*(C-D))
      VECF(A,B,C,D)=(A-B)/DXBF(A,B,C,D)
      IRCAL=2
      IAA=DXBF(XGI,XFI,YGI,YFI)/GSGM
      IF(IAA)17,18,17
```

```
17 IF(IAA-200)1,1,2
18 XG(NG+1)=XG(NG-1) $ YG(NG+1)=YG(NG-1) $ ZG(NG+1)=ZG(NG-1)
   XG(NG+2)=XFI $ YG(NG+2)=YFI $ ZG(NG+2)=ZFI
   XG(NG)=XGI $ YG(NG)=YGI $ ZGH(NG)=ZG(NG)=ZGI
   IAA=2 $ GO TO 8
 2 IF(IAA/2-200)4,4,3
 3 PRINT 1000
   STOP
 1 IB=1 $ GO TO 15
 4 IB=2
15 DO 5 IC=1,IAA
   XG(NG+IC)=XG(NG+IC-1)+(XFI-XGI)*IB*GSGM/DXBF(XGI,XFI,YGI,YFI)
   YG(NG+IC)=YG(NG+IC-1)+(YFI-YGI)*IB*GSGM/DXBF(XGI,XFI,YGI,YFI)
   XL=XG(NG+IC) $ YL=YG(NG+IC) $ CALL ELEV
   K=NG+IC
 5 ZG(NG+IC)=ZE
   XG(NG)=XGI $ YG(NG)=YGI
   ZGH(NG)=ZG(NG)=ZGI
   IF(XG(NG+IAA)-XFI)6,7,6
 6 XG(NG+IAA+1)=XFI $ YG(NG+IAA+1)=YFI $ ZG(NG+IAA+1)=ZFI
   K1=K+1
   IAA=IAA+1 $ GO TO 8
 7 ZG(NG+IAA)=ZFI
 8 CALL CONTOUR(NG+IAA,XG,YG,ZG,ZGH,NG+1)
   GO TO (9,14),IER
 9 DO 10 ID=1,IAA
   IF(ZGI+ID*AC*.0508-ZG(NG+ID)+3.)11,10,10
11 IN=IAA+1-ID $ GO TO 12
10 CONTINUE
   GO TO 3
12 DO 13 LI=1,IN
   YG(NG+IAA+2-LI)=YG(NG+IAA+1-LI)
   XG(NG+IAA+2-LI)=XG(NG+IAA+1-LI)
13 ZG(NG+IAA+2-LI)=ZG(NG+IAA+1-LI)
   XG(NG+IAA+1-IN)=XG(NG+IAA-IN)+IB*GSGM*VECF(YG(NG+IAA-IN),YG(NG+IAA
  1-IN+1),XG(NG+IAA-IN),XG(NG+IAA-IN+1))
```

```
      YG(NG+IAA+1-IN)=YG(NG+IAA-IN)-IB*GSGM*VECF(XG(NG+IAA-IN),XG(NG+IAA
     1-IN+1),YG(NG+IAA-IN),YG(NG+IAA-IN+1))
      XL=XG(NG+IAA+1-IN) $ YL=YG(NG+IAA+1-IN)  $ CALL ELEV
      ZG(NG+IAA+1-IN)=ZE   $ IAA=IAA+1
      GO TO 8
   14 RETURN
      END



      SUBROUTINE PLOT1(XA,YA,XG,YG,XT,YT,IMAX,IGMAX,ITMAX)
C
C    THIS SUBROUTINE CONTROLS GRAPHIC OUTPUT.
C
      COMMON/BCK1/XMAX,XMIN,YMAX,YMIN,ITD
      DIMENSION XA(300),YA(300),XG(300),YG(300),XT(300),YT(300),X(300),
     1          JZ(12),Y(300)
      JZ(1)=8H MOVEMEN
      JZ(2)=8HT OF HEL
      JZ(3)=8HICOPTER
      JZ(4)=8H A/C
      JZ(5)=8H
      JZ(6)=8H
      JZ(7)=8H AND TAN
      JZ(8)=8HKS (JOHN
      JZ(9)=8HSON,VAN
      JZ(10)=8HLEEUWEN)
      JZ(11)=8H
      JZ(12)=8H
      DO 1000 I=1,IMAX
      X(I)=XA(I)-XMIN
 1000 Y(I)=YA(I)-YMIN
      CALL DRAW (IMAX,X,Y,1,0,SCTA,JZ,0,0,0,0,0,0,4,3,1,LAST)
      DO 1001 I=1,IGMAX
```

```
      X(I)=XG(I)-XMIN
 1001 Y(I)=YG(I)-YMIN
      CALL DRAW (IGMAX,X,Y,2,0,ATKA,JZ,0,0,0,0,0,0,4,3,1,LAST)
      DO 1002 I=1,ITMAX
      X(I)=XT(I)-XMIN
 1002 Y(I)=YT(I)-YMIN
      CALL DRAW (ITMAX,X,Y,3,0,TANK,JZ,0,0,0,0,0,0,4,3,1,LAST)
      RETURN
      END



      SUBROUTINE ATTACK
C
C     SUBROUTINE ATTACK CONTROLS THE CONFLICT PORTION OF AMREC.
C
      COMMON/ATT1/XTI,YTI,ZTI,XAI,YAI,ZAI,XGI,YGI,ZGI,TT,TG,TTA,ISFLG,
     1      IGFLG,RNNUM,GSM,TSPD,P2,P1,GSGM,P4,P3,NWPTYP,XA,YA,ZH,XG,YG
     2,ZGH,IMAX,IGMAX,ITMAX,JTYP,TA,FT,RLT,RMAX,RMIN,P,IKILL,TSPDM,TFT,
     ASPT,SPA,SPG,TSV1,TF,TX,GSMX,NS,NG,ZA(300),ZG(300)
     B,HTA,TOF
     A,P5,P6
     C,XT(300),YT(300),ZT(300)
     E,INVC,INV,RC,I9
     4      /RANGE1/N,X,Y,JKT
     5      /ALOS/ITSE,ICAT,DIS,ATR,ISD,IGD
     6      /TGSE/XFI6,YFI6,ZFI6
     7      /ELEV1/PT(7531),XL,YL,ZE,PV(3)
     8      /RTE/IAA
     A      /CONTI/AC,AD,IRCAL,IER
     9      /BCK1/XMAX,XMIN,YMAX,YMIN,ITD
     9      /NUC1/A,B,C,W,PK,TRD,TDAM,TAPDAM
      TYPE INTEGER PO,PK,RNNUM,TA,FT,RLT,P1,P2,P3,P4
      DIMENSION KSS(4),KGG(4),TF(4),TX(4),TY(2),XA(300),YA(300),ZH(300),
```

```
     2            XG(300),YG(300),ZGH(300),JTYP(4),TA(4),FT(4),RLT(4),RMAX
     3            (4),RMIN(4),P(4,25),TFT(4),SPT(25),SPG(25),SPA(25),IT(4)
     4            ,IPAUS(2),INN(2),INC(2),INV(2),INVC(2),TOK(2)
      DXBF(A,B,C,D)=SQRTF((A-B)*(A-B)+(C-D)*(C-D))
      VECF(A,B,C,D)=(A-B)/DXBF(A,B,C,D)
C
C     INITIALIZATION
C
  555 DO 557 LK=1,NWPTYP
      TF(LK)=0
      KSS(LK)=KGG(LK)=0
  557 IT(LK)=0
      IAT=2 $ IA=ISF=IGF=ISD=IGD=AB=CD=0
      INSE=IDI=1
  556 ID=ATT=IJ=INC(1)=INC(2)=INN(1)=INN(2)=0
      TY(1)=TY(2)=IPAUS(1)=IPAUS(2)=TX(1)=TX(2)=TX(3)=TX(4)=0
      IST=K=1
      IRCAL=2
      TOK(1)=TOK(2)=0
      ITP=IPP=0
      ITD=0
      FG=0
      XFI=YFI=0
      IBUG=1
 3050 GO TO(3051,3051,3060,3060),ISFLG
 3051 GO TO(3052,3052,3060,3060),IGFLG
 3052 INSE=2
C
C     TARGET MOVEMENT DETERMINATION
C
 3060 GO TO(552,550,552,550),ISFLG
  550 GO TO (552,551,552,551),IGFLG
  552 IF(IA-2*NWPTYP)553,551,551
  553 ITMOV=1 $ GO TO 919
  551 ITMOV=0
C
```

```
C     INITIAL AIRCRAFT DECISIONS
C
  919 GO TO(600,600,625,625),IGFLG
  600 GO TO (554,554,602,602),ISFLG
  602 IF(ITMOV)603,603,609
  603 DTT=ATT+TT
      DO 605 JX=1,JX $ DTT=DTT+1.
      CALL RANGE(XT,YT,ZT,ITMAX,DTT,TSPD) $ XTI1=X $ YTI1=Y
      XPI=XTI1+ATR*VECF(XGI,XTI,YGI,YTI)
      YPI=YTI1+ATR*VECF(YGI,YTI,XGI,XTI)
      IF(ABSF(DXBF(XGI,XPI,YGI,YPI)-GSGM*JX)-100.)606,606,605
  606 CALL BNDCK(XPI,YPI,2) $ GO TO 608
  605 CONTINUE
  608 XL=XTI1 $ YL=YTI1 $ CALL ELEV $ ZTI1=ZE
      P61=DXBF(XPI,XTI1,YPI,YTI1)
      CALL TGSE(XTI1,YTI1,P61,XPI,YPI,XFI,YFI,ZFI,ZTI1,1)
      XGSV=XFI6 $ YGSV=YFI6 $ ZGSV=MAX1F(ZFI6,ZFI)
  630 IF(ISD-2)6300,6140,6140
 6300 XSI=XTI1-ATR*VECF(YFI,YTI1,XFI,XTI1) $ YSI=YTI1+ATR*VECF(XFI,XTI1,
     1YFI,YTI1) $ XST1=XTI1+ATR*VECF(YFI,YTI1,XFI,XTI1) $ YST1=YTI1-ATR*
     2VECF(XFI,XTI1,YFI,YTI1)
       GO TO 615
  635 CALL TGSE(XTI,YTI,P6,XGI,YGI,XFI,YFI,ZFI,ZTI,2)
      CALL RTE(NG,XG,YG,ZG,ZGH,XGI,YGI,ZGI,XFI,YFI,ZFI,GSGM)
      IGMAX=NG+IAA
  609 CALL TGSE(XTI,YTI,P6,XGI,YGI,XFI,YFI,ZFI,ZTI,1)
      XGSV=XFI6 $ YGSV=YFI6 $ ZGSV=ZFI6
      IF(ISD-2)6090,6140,6140
 6090 XSI=XTI-ATR*VECF(YFI,YTI,XFI,XTI) $ YSI=YTI+ATR*VECF(XFI,XTI,YFI,Y
     1TI) $ XST1=XTI+ATR*VECF(YFI,YTI,XFI,XTI) $ YST1=YTI-ATR*VECF(XFI,X
     2TI,YFI,YTI)
      XTI1=XTI $ YTI1=YTI $ ZTI1=ZTI
  615 IF(DXBF(XAI,XSI,YAI,YSI)-DXBF(XAI,XST1,YAI,YST1))612,612,611
  611 XSI=XST1 $ YSI=YST1
  612 XL=XSI $ YL=YSI $ CALL ELEV $ ZSI=ZE+50
      CALL BNDCK(XSI,YSI,1)
```

```
618 CALL ALOS(XFI,YFI,ZFI,XSI,YSI,ZSI) $ ISTF=ITSE
    CALL ALOS (XTI1,YTI1,ZTI1,XSI,YSI,ZSI) $ ISTT=ITSE
    IF(ISTF+ISTT-4)613,614,613
613 IF(ISTT-2) 621,617,621
617 ATR1=ATR $ ATR=P61=DXBF(XSI,XFI,YSI,YFI)
    CALL TGSE(XFI,YFI,P61, XSI,YSI,XSTI,YSTI,ZSTI,ZFI,1)
    CALL ALOS (XTI1,YTI1,ZTI1,XSTI,YSTI,ZSTI)
    ATR=ATR1
    IF(ITSE-2) 620,619,620
619 XSI=XSTI $ YSI=YSTI $ ZSI=ZSTI $ GO TO 614
620 GO TO (624,624,614,614),IGFLG
624 XFI=XGSV $ YFI=YGSV $ ZFI=ZGSV $ GO TO 614
621 ATR1=ATR $ ATR=P61=DXBF(XSI,XTI1,YSI,YTI1)
    CALL TGSE(XTI1,YTI1,P61,XSI,YSI,XSTI,YSTI,ZSTI,ZTI1,1)
    CALL ALOS(XFI,YFI,ZFI,XSTI,YSTI,ZSTI)
    ATR=ATR1
    IF(ITSE-2)623,619,623
623 XSI=XSTI $ YSI=YSTI $ ZSI=ZSTI $ GO TO 620
625 IF(ABSF(P6-ATR)-500.)601,601,602
601 IF(ITMOV)627,627,626
627 IDTT=ATT+HTA+P6/TOF
C
C     TOF=TIME OF FLIGHT OF MISSLE
C     HTA=ACQUISITION TIME
C
    DTT=TT
    DO 629 JT=1,IDTT
    DTT=DTT+1.
    CALL RANGE(XT,YT,ZT,ITMAX,DTT,TSPD) $ XL=X $ YL=Y $ CALL ELEV
    CALL ALOS(XL,YL,ZE,XGI,YGI,ZGI)
    IF(ITSE-2)603,629,603
629 CONTINUE
    XFI=XGI $ YFI=YGI $ ZFI=ZGI $ GO TO 630
626 IF(IGF-1)634,634,635
634 XFI=XGI $ YFI=YGI $ ZFI=ZGI $ GO TO 554
614 XSE=XA(IMAX) $ YSE=YA(IMAX) $ ZSE=ZH(IMAX)
```

```
 6140 XGE=XG(IGMAX) $ YGE=YG(IGMAX) $ ZGE=ZGH(IGMAX)
      CALL RTE(NG,XG,YG,ZG,ZGH,XGI,YGI,ZGI,XFI,YFI,ZFI,GSGM)
      NA=NG+IAA $ CALLRTE(NA,XG,YG,ZG,ZGH,XG(NA),YG(NA),ZGH(NA),XTI1,YT
     1I1,ZTI1+100,GSGM)
      NA=NA+IAA $ CALL RTE(NA,XG,YG,ZG,ZGH,XG(NA),YG(NA),ZGH(NA),XGE,YGE
     1,ZGE,GSGM) $ IGMAX=NA+IAA
      IF(ISD-2)6041,554,554
 6041 CALL RTE(NS,XA,YA,ZA,ZH,XAI,YAI,ZAI,XSI,YSI,ZSI,GSM)
      XL=XTO=2*XTI1-XFI $ YL=YTO=2*YTI1-YFI $ CALL ELEV $ ZTO=ZE+30
      NA=NS+IAA $ CALL RTE(NA,XA,YA,ZA,ZH,XA(NA),YA(NA),ZH(NA),XTO,YTO
     1,ZTO,GSM)
      NA=NA+IAA $ CALL RTE(NA,XA,YA,ZA,ZH,XA(NA),YA(NA),ZH(NA),XSE,YSE,Z
     1SE,GSM) $ IMAX=NA+IAA
      GSM1=DXBF(XAI,XSI,YAI,YSI)*GSGM/DXBF(XGI,XFI,YGI,YFI)
      GSVM=GSM
      GSM=MIN1F(GSM1,GSMX)
  554 DO 100 JJ=K,JJ
C
C     ELEMENT POSITION DETERMINATION AND SCOUT POP UP CONTROL.
C
       ATT=ATT+1
      IF(ITMOV-1)950,949,949
  950 TT=TT+1 $ CALL RANGE(XT,YT,ZT,ITMAX,TT,TSPD)
      NT=N+1
      XL=XTI=X $ YL=YTI=Y $ CALL ELEV $ ZTI=ZE
  949 IF(ISD-1)920,920,921
  920 DTT=ATT+TSV1
      CALL RANGE(XA,YA,ZH,IMAX,DTT,GSM)  $ XAI=X $ YAI=Y $ ZAI=ZH(N)+(XA
     1I-XA(N))*(ZH(N+1)-ZH(N))/(XA(N+1)-XA(N)) $ CALL ALOS (XAI,YAI,ZAI,
     2XTI,YTI,ZTI) $ P1=ICAT $ P5=DIS $ P2=ITSE $ NS=N+1
      IF(P2-2)922 ,3062,3062
 3062 IF(INSE-2)3063,3064,3064
 3064 ATT=ATT-1 $ INSE=1 $ GO TO 3060
 3063 IF(IST-1)922,922,9230
 9230 IF(IPP)400,400,401
  400 IPP=IPP+1 $  ZAI=ZAI+RC*.00508 $ ZAII=ZAI $ CALL ALOS(XTI,YTI,ZTI,
```

```
     1XAI,YAI,ZAI) $ P1=ICAT $ P2=ITSE $ P5=DIS $ IF(P2-2)922,402,402
  402 IF(INSE-2)3065,3064,3064
 3065 DO 403 IY=1,NS
      XA(IMAX+2-IY)=XA(IMAX+1-IY) $ YA(IMAX+2-IY)=YA(IMAX+1-IY)
  403 ZA(IMAX+2-IY)=ZA(IMAX+1-IY)
      XL=XA(NS)=XAI $ YL=YA(NS)=YAI
      CALL FLEV $ ZA(NS)=ZE $ IMAX=IMAX+1 $ CALL CONTOUR(IMAX,XA,YA,ZA,Z
     1H,NS) $ IF(IPP-I9)407,408,408
  407 IPP=0 $ GO TO (404,404,922,922),ISFLG
  404 ISFLG=ISFLG+2 $ GO TO 922
  401 IF(IPP-I9)405,405,406
  405 IPP=IPP+1 $ ZAI=ZAII+.00508*IPP $ CALL ALOS(XTI,YTI,ZTI,XAI,YAI,ZA
     1I) $ ZAII=ZAI $ P1=ICAT $ P2=ITSE $ P5=DIS $ IF(P2-2)922,402,402
  406 ZAI=ZAII+RC*.00508*IPP $ CALL ALOS(XTI,YTI,ZTI,XAI,YAI,ZAI) $ P1=I
     1CAT $ IPP=0
      P2=ITSE $ P5=DIS $ IF(P2-2)3065,4020,3065
 4020 GO TO(4021,4021,402,402),ISFLG
 4021 ISFLG=ISFLG+2 $ GO TO 402
  408 IPP=0 $ GO TO 922
  921 P2=1 $ ISFLG=2 $ XAI=YAI=ZAI=0
  922 IF(IGD-1)923,923,924
  923 DTT=ATT+TG $
     3CALL RANGE(XG,YG,ZGH,IGMAX,DTT,GSGM) $ XGI
     4=X $ YGI=Y $ ZGI=ZGH(N)+( XGI-XG(N))*(ZGH(N+1)-ZGH(N))/(XG(N+1)-XG
     5(N)) $ CALL ALOS(XGI,YGI,ZGI,XTI,YTI,ZTI) $ P3=ICAT $ P6=DIS $
     6P4=ITSE $ NG=N+1
      IF(P4-2)2345,3066,3066
 3066 IF(INSE-2)925,3064,3064
  924 P4=1 $ IGFLG=2 $XGI=YGI=ZGI=0 $ GO TO 925
 2345 EF=ATT
  925 PRINT 4747,TSV1,ATT,XAI,YAI,ZAI,XGI,YGI,ZGI,XTI,YTI,ZTI,P5,P6,P2,P
     14
      GO TO(999,92),JKT
C
C     TARGET GUNFIRE CONTROL.
C
```

```
   92 IF(P2-2)20,2,20
    2 GO TO (3,9,3,9),ISFLG
    3 IF(IPAUS(1))703,703,704
  704 IPAUS(1)=0 $ AB=ATT
  703 ID=3 $ TY(1)=0
   70 IF(P4-2)11,4,11
    4 GO TO (270,14,270,14),IGFLG
  270 IF(IPAUS(2))705,705,706
  706 IPAUS(2)=0 $ CD=ATT
  705 TY(2)=0    $ GO TO 30
   20 GO TO (8,6,8,6),ISFLG
    6 ID=1 $ GO TO 70
    8 IF=1 $ GO TO 16
    9 IF(RANF(-1.)-SPT(P1))10,10,6
   10 ISFLG=ISFLG-1
      AB=ATT
      IF(IA-2*NWPTYP)647,30,30
  647 ITMOV=1 $ GO TO 704
   11 GO TO (13,12,13,12),IGFLG
   12 ID=ID+1 $ GO TO 30
   13 IF(TOK(1)+TOK(2)) 5051,5051,5052
 5051 IF=2 $ GO TO 16
 5052 TT=TT-1 $ TT=TT*TSPD/TSPDM $ TSPD=TSPDM $ ITMOV=0 $ IGFLG=IGFLG+1
      GO TO 950
   14 IF(RANF(-1.)-SPT(P3)) 140,140,12
  140 IGFLG=IGFLG-1
      IF(IA-2*NWPTYP)648,30,30
  648 ITMOV=1 $ GO TO 706
   16 IF(ATT-TY(IF))981,981,29
  281 IPAUS(IF)=0
  981 IF(IF-1)6,6,12
  160 IP=0
   17 IP=IP+1
      ATT=ATT+1
      IF(IF-1)21,18,21
   18 DTT=ATT+TSV1
```

```
    1 CALL RANGE(XA,YA,ZH,IMAX,DTT,GSM)
       XAI=X $  YAI= Y $ ZAI=(ZH(N)+( XAI-XA(N))*(ZH(N+1)-ZH(N))/(XA(N+
     11)-XA(N)))
      GO TO(999,93),JKT
   93 CALL ALOS(XAI,YAI,ZAI,XTI,YTI,ZTI) $ P1=ICAT
      IF(ITSF-2)19,25,19
   21 DTT=ATT+TG
    7 CALL RANGE(XG,YG,ZGH,IGMAX,DTT,GSGM) $ XGI=X $ YGI=Y $ ZGI=
     1ZGH(N)+(XGI-XG(N))*(ZGH(N+1)-ZGH(N))/(XG(N+1)-XG(N))
      GO TO(999,94),JKT
   94 CALL ALOS(XGI,YGI,ZGI,XTI,YTI,ZTI) $ P3=ICAT
      IF(ITSF-2)19,26,19
   19 IF(IP-15)17,17,22
   22 GO TO (283,284),IF
  283 IPAUS(IF)=0 $GO TO 27
  280 IPAUS(IF)=1
   27 ATT=ATT-IP $ GO TO 6
  284 IPAUS(IF)=0 $GO TO 28
  282 IPAUS(IF)=1
   28 ATT=ATT-IP $ GO TO 12
   25 TY(IF)=ATT $ GO TO 280
   26 TY(IF)=ATT $ GO TO 282
   29 IF(TY(IF))160,160,281
C
C     AIRCRAFT FIRE CONTROL AND SUBSEQUENT ATTACK DECISIONS.
C
   30 IF(IGD-2)890,659,659
  890 IF(P4-2)409,410,410
  410 GO TO(424,424,427,427),IGFLG
  424 IGFLG=IGFLG+2
  427 IF(P6-500)951,952,952
  952 IF(INC(1).LT.INVC(1).OR.INC(2).LT.INVC(2))953,951
  951 IF(P6-100)445,954,954
  954 IF(INN(1).LT.INV(1).OR.INN(2).LT.INV(2))955,445
  955 INUC=2 $ GO TO 428
  953 INUC=1
```

```
428 IF(ATT-HTA-EF)659,429,429
429 IF(IDI-2)430,430,431
430 IF(TOK(1))804,804,431
804 IF(INUC-1)805,805,806
805 IF(INC(1)-INVC(1)) 432,807,807
807 IF(INN(1)-INV(1)) 808,809,809
808 INUC=2$ GO TO 432
809 IF(IGD)810,810,811
810 IF(INC(2)+INN(2)-INVC(2)-INV(2))437,811,811
811 CALL TGSE(XTI,YTI,P6,XGI,YGI,XGTI,YGTI,ZGTI,ZTI,2)
    CALL RTE (NG,XG,YG,ZG,ZGH,XGI,YGI,ZGI,XGTI,YGTI,ZGTI,GSGM)
    XGE=XG(IGMAX) $ YGE=YG(IGMAX) $ ZGE=ZGH(IGMAX)
    NA=NG+IAA
    CALL RTE(NA,XG,YG,ZG,ZGH,XG(NA),YG(NA),ZGH(NA),XGE,YGE,ZGE,GSGM)
    IGMAX=NA+IAA $ GO TO 661
806 IF(INN(1)-INV(1)) 432,812,812
812 IF(INC(1)-INVC(1))813,809,809
813 IF(IGD)814,814,445
814 IF(INC(2)+INN(2)-INVC(2)-INV(2))437, 445,445
432 TOK(1)=ATT+P6/TOF-1.
    IF(ITMOV)433,433,434
434 XK1=XTI $ YK1=YTI $ GO TO 825
825 IF(INUC-1)826,826,827
826 INC(1)=INC(1)+1 $ GO TO 437
827 INN(1)=INN(1)+1 $ GO TO 437
433 CALL RANGE(XT,YT,ZT,ITMAX,TOK(1)+TT,TSPD) $ XL=X $ YL=Y $CALL ELEV
    CALL ALOS(X,Y,ZE,XGI,YGI,ZGI) $ IF(ITSE-1) 435,435,436
436 XK1=X $ YK1=Y   $ GO TO 825
435 TOK(1)=0 $ GO TO 437
437 IF(IDI-1)431,659,659
431 IF(TOK(2)) 815,815,659
815 IF(INUC-1) 816,816,817
816 IF(INC(2)-INVC(2)) 438,818,818
818 IF(INN(2)-INV(2)) 819,820,820
819 INUC=2 $ GO TO 438
820 IF(IGD)821,821,811
```

```
 821 IF(INC(1)+INN(1)-INV(1)-INVC(1))659,811,811
 817 IF(INN(2)-INV(2))438,822,822
 822 IF(INC(2)-INVC(2)) 823,824,824
 823 IF(IGD)8250,8250,445
8250 IF(TOK(1))659,445,659
 824 IF(IGD)821,821,811
 438 TOK(2)=ATT+P6/TOF-1
     IF(ITMOV)439,439,440
 440 XK2=XTI $ YK2=YTI $ GO TO 828
 828 IF(INUC-1)829,829,830
 829 INC(2)=INC(2)+1 $ GO TO 659
 830 INN(2)=INN(2)+1 $ GO TO 659
 439 CALL RANGE(XT,YT,ZT,ITMAX,TOK(2)+TT,TSPD) $XL=X $YL=Y $CALL ELEV
     CALL ALOS(X,Y,ZE,XGI,YGI,XGI) $ IF(ITSF-1)441,441,442
 441 TOK(2)=0 $ GO TO 659
 442 XK2=X $ YK2=Y $ GO TO 659
 409 GO TO (659,659,412,412),IGFLG
 412 IGFLG=IGFLG-2
 445 CONTINUE
 659 IF(ISD-1)801,801,661
 801 IF(P2-2)4590,450,450
4590 P61=DXBF(XAI,XTI1,YAI,YTI1)
     IF(P61-500.)661,661,452
 452 IF(IGD-1)454,454,661
 454 IF(DXBF(XGI,XFI,YGI,YFI)-150.)458,458,661
 458 GO TO(4440,4440,661,661),IGFLG
4440 IF(TOK(1)+TOK(2))457,457,661
 457 IST=2 $ GO TO 661
 450 IF(IST-1)4591,4591,459
 459 IPP=19
4591 P61=DXBF(XAI,XTI,YAI,YTI)
     IF(ISF-1)451,451,661
 451 GO TO(650,650,461,461),IGFLG
 650 IF(IGD-1)443,443,461
 443 IF(TOK(1)+TOK(2))444,444,461
 444 IF(DXBF(XGI,XFI,YGI,YFI)-50.)4445,4445,461
```

```
4445 IF(DXBF(XTI,XTI1,YTI,YTI1)-50.)661,661,4451
4451 IF(ITMOV)446,446,4710
 446 DO 447,IVP=1,IVP
     DTT=ATT+TT+.5  $ CALL RANGE(XT,YT,ZT,ITMAX,DTT,TSPD)
     XTI1=X $ YTI1=Y
     XPI=XTI1+ATR*VECF(XGI,XTI,YGI,YTI)
     YPI=YTI1+ATR*VECF(YGI,YTI,XGI,XTI) $ IF(ABSF(DXBF(XGI,XPI,YGI,YPI)
    1-GSGM*IVP*.5)-10)448,448,447
 447 CONTINUE
 448 CALL BNDCK(XPI,YPI,2)
     XL=XTI1 $ YL=YTI1 $ CALL ELEV $ ZTI1=ZE
     GO TO 4720
4710 XTI1=XTI $ YTI1=YTI $ ZTI1=ZTI
     XPI=XTI+ATR*VECF(XGI,XTI,YGI,YTI)
     YPI=YTI+ATR*VECF(YGI,YTI,XGI,XTI)
     CALL TGSE(XTI,YTI,ATR,XPI,YPI,XFI,YFI,ZFI,ZTI,1)
4720 XGE=XG(IGMAX) $ YGE=YG(IGMAX) $ ZGE=ZGH(IGMAX)
     CALL RTE(NG,XG,YG,ZG,ZGH,XGI,YGI,ZGI,XFI,YFI,ZFI,GSGM)
     NA=NG+IAA
     CALL RTE(NA,XG,YG,ZG,ZGH,XG(NA),YG(NA),ZGH(NA),XTI1,YTI1,ZTI1,GSGM
    1) $ NA=NA+IAA
     CALL RTE(NA,XG,YG,ZG,ZGH,XG(NA),YG(NA),ZGH(NA),XGE,YGE,ZGE,GSGM)
     IGMAX=NA+IAA $ IF(IGMAX-300)661,661,449
 461 ATR1=ATR $ ATR=P61
     CALL TGSE(XTI,YTI,P61,XAI,YAI,XSTI,YSTI,ZSTI,ZTI,2)
     IMAK=IMAX-NS+1
     DO 4730 I=1,IMAK
     XA(IMAX+3-I)=XA(IMAX+1-I)
     YA(IMAX+3-I)=YA(IMAX+1-I)
     ZA(IMAX+3-I)=ZA(IMAX+1-I)
4730 ZH(IMAX+3-I)=ZH(IMAX+1-I)
     XL=XA(NS)=XAI $ YL=YA(NS)=YAI $ ZH(NS)=ZAI $ CALL ELEV $ ZA(NS)=ZE
     XL=XA(NS+1)=XSTI $ YL=YA(NS+1)=YSTI $ ZH(NS)=ZSTI $ CALL ELEV
     ZA(NS+1)=ZE $ IMAX=IMAX+2 $ CALL CONTOUR(IMAX,XA,YA,ZA,ZH,NS)
     ATR=ATR1
     GO TO 661
```

```
  449 PRINT 9010 $ STOP
 9010 FORMAT(///1X,47HMATRIX DIMENSION EXCEEDED. THIS RUN TERMINATED.)
C
C      ARMOR DAMAGE ASSESSMENT
C
  661 IF(TOK(1)) 463,463,464
  464 IF(ATT-TOK(1))463,465,465
  465 CALL NUC(INUC,XT(NT),YT(NT),XT(NT-1),YT(NT-1),ITMOV,XK1,YK1)
      I1=TDAM $ I2=TAPDAM
      IF(TDAM+TAPDAM)467,467,7601
 7601 IF(JTYP(1)-TDAM)7500,7501,7501
 7500 I2=TAPDAM=TDAM+TAPDAM-JTYP(1) $ I1=JTYP(1) $ JTYP(1)=JTYP(2)=0
      GO TO 7603
 7604 I2=JTYP(3) $ JTYP(3)=JTYP(4)=0$IF(JTYP(3)+JTYP(1))7602,7602,466
 7602 ITP=5 $ GO TO 466
 7501 JTYP(1)=JTYP(2)=JTYP(2)-TDAM
 7603 IF(JTYP(3)-TAPDAM)7604,7604,7605
 7605 JTYP(3)=JTYP(4)=JTYP(4)-TAPDAM
  466 TOK(1)=0 $ ITD=100 $ GO TO 463
  467 TOK(1)=0 $ ITD=10
  463 IF(TOK(2))469,469,468
  468 IF(ATT-TOK(2))469,470,470
  470 CALL NUC(INUC,XT(NT),YT(NT),XT(NT-1),YT(NT-1),ITMOV,XK2,YK2)
      I3=TDAM $ I4= TAPDAM
      IF(TDAM+TAPDAM)472,472,7801
 7801 IF(JTYP(1)-TDAM)7700,7701,7701
 7700 I4=TAPDAM=TDAM+TAPDAM-JTYP(1) $ I3=JTYP(1) $ JTYP(1)=JTYP(2)=0
      GO TO 7803
 7701 JTYP(1)=JTYP(2)=JTYP(2)-TDAM
 7803 IF(JTYP(3)-TAPDAM)7804,7804,7805
 7804 I4=JTYP(3) $ JTYP(3)=JTYP(4)=0 $ IF(JTYP(3)+JTYP(1))7802,7802,471
 7802 ITP=5 $ GO TO 471
 7805 JTYP(3)=JTYP(4)=JTYP(4)-TAPDAM
  471 TOK(2)=0 $ ITD=ITD+200 $ GO TO 469
  472 TOK(2)=0 $ ITD=ITD+20
  469 ITD=ITD/100+1
```

```
C
C      AIRCRAFT DAMAGE ASSESSMENT
C
 3000 IXP=IKILL $ KS=KG=0 $ IGF=ISF=1 $ GO TO(31,99,510,38),ID
   31 IF=2 $ GO TO 37
   38 IF=1
   37 DO 50 J=1,NWPTYP
      IF(JTYP(J))4400,4400,4401
 4400 KSS(J)=KGG(J)=0 $ GO TO 50
 4401 GO TO(32,41),IF
   32 IF(ATT-TA(J)-AB)44,33,33
   33 IF(ATT-1.-TX(J))44,34,34
   34 IF(TF(J)+1.-FT(J))35,35,36
   36 TX(J)=ATT+RLT(J)+TA(J)/2-1 $ TF(J)=0 $ IT(J)=IT(J)+1 $ KSS(J)=KGG(
     1J)=0 $ GO TO 50
   35 IF(IT(J)*FT(J)+TF(J)+1.-TFT(J))370,370,380
  380 KSS(J)=KGG(J)=2 $ GO TO 50
  370 IF(P5-RMAX(J))39,39,44
   39 IF(RMIN(J)-P5)40,40,44
   40 KSS(J)=1 $ KS=KS+10**J $ GO TO 50
   44 KSS(J)=0 $ GO TO 50
   41 IF(ATT-TA(J)-CD)46,42,42
   42 IF(ATT-1.-TX(J))46,43,43
   43 IF(TF(J)+1.-FT(J))45,45,36
   45 IF(IT(J)*FT(J)+TF(J)+1.-TFT(J))47,47,380
   47 IF(P6-RMAX(J))48,48,46
   48 IF(RMIN(J)-P6)49,49,46
   49 KGG(J)=1 $ KG=KG+10**J $ GO TO 50
   46 KGG(J)=0
   50 CONTINUE
   53 APS=APG=1. $ DO 54 JP=1,NWPTYP $ IF(KSS(JP)-2)540,54,54
  540 APS=APS*(1.-KSS(JP)*P(JP,P1))**JTYP(JP)
      APG=APG*(1.-KGG(JP)*P(JP,P3))**JTYP(JP)
   54 CONTINUE
      PRINT 4748,APS,APG
      GO TO (56,55,51,55),ID
```

```
 55 IF(1.-APS-.15)57,58,58
 57 ISF=1 $ GO TO 99
 58 IF(RANF(-1.)-1.+APS)61,61,59
 59 ISF=2 $ GO TO 99
 61 ISF=3 $ GO TO 1098
 56 IF(1.-APG-.15)63,64,64
 63 IGF=1 $GO TO 99
 64 IF(RANF(-1.)-1.+APG)66,66,65
 65 IGF=2 $ GO TO 99
 66 IGF=3 $ GO TO 1097
510 PO=0
 51 PO=PO+1
     GO TO (38,31,52),PO
 52 IJ=0
    GO TO (67,72,73,76),IKILL
 67 IF(KS)55,68,55
 68 IJ=1
 72 IF(KG)56,69,56
 69 IF(IJ)71,71,1099
 71 IF(KS)55,1099,55
1099 ISF=IGF=1 $ GO TO 99
 73 IF(APG-APS) 560,560,559
559 IKILL=IXP $ GO TO 55
560 IKILL=IXP $ GO TO 56
 76 APS=APG=1 $ DO 771 PK=1,NWPTYP
    IF(P(PK,P1)-P(PK,P3))75,74,74
 74 KGG(PK)=0 $ GO TO 761
 75 KSS(PK)=0
761 IF(KGG(PK).EQ.2.OR.KSS(PK).EQ.2)771,760
760 APG=APG*(1.-KGG(PK)*P(PK,P3))**JTYP(PK)
 77 APS=APS*(1.-KSS(PK)*P(PK,P1))**JTYP(PK)
771 CONTINUE
    IF(1.-APS-.15)770,78,78
770 IKILL=3 $ GO TO 510
 78 IF(RANF(-1.)-1.+APS)80,80,79
 79 ISF=2 $ GO TO 83
```

```
   80 ISF=3 $ GO TO 83
   83 IF(1.-APG-.15)84,85,85
   84 IKILL=3 $ GO TO 510
   85 IF(RANF(-1.)-1.+APG)87,87,86
   86 IGF=2 $ GO TO 99
   87 IGF=3
 1097 IGD=IGD+1 $ GO TO 99
 1098 ISD=ISD+1
C
C      INTERACTION EFFECTS.
C
   99 GO TO(473,474,474,474),ITD
  473 GO TO(477,477,476),IGF
  477 GO TO(1000,1000,479),ISF
  479 CALL PRINT1(ATT,0,KS,KG,1,0,0) $ GO TO 1000
  476 GO TO(932,936),IGD
  932 IF(RANF(-1.)-RANF(-1.))934,934,935
  934 IDI=2 $ GO TO 936
  935 IDI=3
  936 CALL PRINT1(ATT,0,KS,KG,2,0,0) $ GO TO 1000
  474 GO TO(940,940,943),IGF
  940 GO TO(942,942,944),ISF
  943 CALL PRINT1(ATT,0,KS,KG,2,0,0)
  942 CALL PRINT1(ATT,INUC,I1+I3,I2+I4,3,ITD,0) $ GO TO 1000
  944 CALL PRINT1(ATT,0,KS,KG,1,0,0) $ GO TO 942
C
C      MISCELLANEOUS
C
 1000 I1=I2=I3=I4=0$ IF(ITP-5)7710,7800,7800
 7710 GO TO(101,102,103,103),ID
  101 GO TO(104,105,105),IGF
  104 IA=0
      DO 201 JM=1,NWPTYP
  201 IA=IA+KGG(JM)
      IF(IA-2*NWPTYP)97,90,97
  105 DO 202 JN=1,NWPTYP
```

```
      IF(KSS(JN)-2)200,202,200
  200 TF(JN)=TF(JN)+KGG(JN)
  202 CONTINUE $ GO TO 97
  102 JXF=IPAUS(1)+IPAUS(2)+1
      GO TO (302,97,97),JXF
  302 GO TO (106,107,106,107),ISFLG
  106 ISFLG=ISFLG+1 $ GO TO 90
  107 GO TO(108,91,108,91),IGFLG
  108 IGFLG=IGFLG+1 $ GO TO 90
  103 GO TO(109,110,110),ISF
  109 IA=0
      DO 203 JO=1,NWPTYP
  203 IA=IA+KGG(JO)
      IF(IA-2*NWPTYP)111,90,111
  110 DO 204 JP=1,NWPTYP
      IF(KSS(JP)-2)205,204,205
  205 TF(JP)=TF(JP)+KSS(JP)
  204 CONTINUE
  111 IF(ID-3)97,101,97
   97 GO TO(636,636,333,333),ISFLG
  636 IF(P2-2)643,637,637
  637 IF(ISF-2)639,642,641
  639 IF(RANF(-1.)-SPA(P1))642,642,643
  642 ISFLG=ISFLG+2 $ IBUG=3
  643 GO TO(638,638,3333,3333),IGFLG
 3333 IBUG=0 $ GO TO 333
  641 IF(ISD-1)642,642,643
  638 IF(P4-2)333,640,640
  640 IF(IGF-2)644,646,645
  644 IF(RANF(-1.)-SPG(P3))646,646,333
  646 IGFLG=IGFLG+2
      IBUG=3 $ GO TO 333
  645 IF(IGD-1)646,646,100
   90 TT=TT*TSPD/TSPDM $ TSPD=TSPDM $ GO TO 91
   91 ITMOV=0
  333 PRINT 9999,ISF,IGF,IF,ID,IPAUS,IA,KSS,KGG,TF,FT,IT,TFT
```

```
      PRINT 9998,TX,TY,ISFLG,IGFLG
      PRINT 9997,TOK,INUC,ITD,INVC,INV,ITMOV,ISD,IGD,JTYP
      IF(IBUG-3)100,909,100
  909 K=JJ+1 $ GO TO 625
  100 CONTINUE
 7800 CALL PRINT1(0,0,0,0,0,0,3) $ JKT=1
  999 PRINT 9999,ISF,IGF,IF,ID,IPAUS,IA,KSS,KGG,TF,FT,IT,TFT
      PRINT 9997,TOK,INUC,ITD,INVC,INV,ITMOV,ISD,IGD,JTYP
 9997 FORMAT(2X,6HTOK = ,2(F10.5,5X),7HINUC = ,I8,5X,6HITD = ,I10//1X,7H
     1INVC = ,2(I10,5X),6HINV = ,2(I10,5X)//1X,7HITMV = ,I10,5X,6HISD =
     2I10,5X,6HIGD = ,I10//1X,7HJTYP = ,4(I10,5X)//)
 9998 FORMAT(3X,5HTX = ,4(F10.5,5X)//3X,5HTY = ,2(F10.5,5X)//1X,8HISFLG
     A= ,I
     110,5X,8HIGFLG = ,I10//)
 9999 FORMAT(2X,6HISF = ,I10,5X,6HIGF = ,I10,5X,6H IF = ,I10,5X,6H ID =
     1,I10//1X,7HIPAS = ,2(I10,5X),5HIA = ,I10//2X,6HKSS = ,4(I10,5X)//2
     AX,
     26HKGG = ,4(I10,5X)//3X,5HTF = ,4(F10.5,5X)//3X,5HFT = ,4(I10,5X)
     3//3X,5HIT = ,4(I10,5X)//2X,6HTFT = ,4(F10.0,5X)//)
 4747 FORMAT(//1X,7HTSV1 = ,F10.6,5X,6HATT = ,F10.6//2X6HXAI = ,F10.3,5X
     1,6HYAI = ,F10.3,5X,6HZAI = ,F10.3//2X6HXGI = ,F10.3,5X,6HYGI = ,F1
     20.3,5X,6HZGI = ,F10.3//2X,6HXTI = ,F10.3,5X,6HYTI = ,F10.3,5X,6HZT
     3I = ,F10.3//23X,8HSCOUT = ,F10.4/5X,18HDIST. TO TGT FROM /23X,8HA-
     4G   = ,F10.4//23X,8HSCOUT = ,I10/11X,12HLOS, TGT TO /23X,8HA-G   =
     5 ,I10//)
 4748 FORMAT(1X,25HPROB OF SCOUT SURVIVAL = ,F10.5//3X,23HPROB OF A-G SU
     1RVIVAL = ,F10.5//)
      PRINT 9998,TX,TY,ISFLG,IGFLG
      RETURN
      END


      SUBROUTINE NUC(JNUC,XTA,YTA,XTB,YTB,ITMOV,XK,YK)
```

```
      COMMON/NUC1/A,B,C,W,PK,TRD,TDAM,TAPDAM
     1       /ATT1/XT,YT,ZT,NN(3),XG,YG,ZG,LN(7),TSPD,P(2),GSGM,AF(2626),
     2             P6,IDN(906)
      SQ(A,B)=SQRTF(A*A+B*B)
      SQB(F,G,H)=SQRTF(F*F+G*G+H*H)
      ASF(B,C,D)=ASINF(B*C/D)
      XTSA=XTA $ YTSA=YTA $ XTSB=XTB $ YTSB=YTB
      TDAM=TAPDAM=0 $ KIND=IKIND=JN=0
      GO TO (889,961),JNUC
C
C     JNUC=1   NUC WEAPON
C         =2 CONVENTIONAL WEAPON WITH PK
C
  889 DISTTG=P6
      J=IND=0  $ DX=DY=0.
      DX=XT-XG $ DY=YT-YG
      IF(ITMOV)900,900,917
C
C     COMPUTE RELATIVE MOTION
C
  900 J=J+1
  906 IF(DX)901,902,902
  901 DX=ABSF(DX) $ IND=IND+2 $GO TO 903
  902 IND=IND+1
  903 IF(DY)904,905,905
  904 DY=ABSF(DY)  $ IND=IND+2
  905 THETA=ATANF(DY/DX)
      GO TO (907,908,909,910),IND
  908 THETA=3.1416-THETA  $ GO TO 907
  909 THETA=6.2832-THETA  $ GO TO 907
  910 THETA= 3.1416+THETA
  907 GO TO(911,912,1011,1012,1012),J
  911 THETAG=THETA
      IND=0 $ DX=DY=0.
      DX=XTA-XT $ DY=YTA-YT $ GO TO 900
  912 THETAT=THETA
```

```
      IF(THETAG-THETAT)913,914,914
  913 ETA = 6.2832 - THETAT + THETAG    $ GO TO 915
  914 ETA = THETAG - THETAT
  915 RFLSPD=SQRTF(TSPD*TSPD+GSGM*GSCM-2.0*TSPD*GSGM*COSF(ETA))
  916 RLTHETA=SINF(TSPD/RELSPD*SINF(ETA))
      XSPD = ABSF(RLTHETA*RELSPD)  $ GO TO 918
  917 XSPD = 0
  918 CEP=DISTTG/200.+XSPD/(1.6*DISTTG/100.)
      SIGMA = CFP*.847
      SRF=SPHIZ=0.
      PHIZ=359.999*RANF(-1.)
      SUM=0.0
      DO 919 I=1,12
  919 SUM = SUM + RANF(-1.)
      XS = SUM-6.0
      RO = ABSF(XS*SIGMA+0.0)
      IF(XK-XT)950,951,950
  950 XT2=XK $ GO TO 952
  951 XT2=XT
  952 IF(YK-YT)953,954,953
  953 YT2=YK $ GO TO 920
  954 YT2=YT
  920 XB = XT2+RO*COSF(PHIZ/57.295)
      YB = YT2+RO*SINF(PHIZ/57.295)
      ZB = ZT
      DISTGB=SQB((XG-XB),(YG-YB),(ZG-ZB))
C
C     SCALING LAW CONVERSION
C     COEFFICIENTS  A,B,C CAN BE FOUND IN DASA PUBS
C
      RIKT= DISTGB/(1.6*W)**.333*3.28003
      RF = LOGF(1000./RIKT)
      DP= EXPF(A+B*RF+C*RF**2)
      DISTTB=SQ((XT-XB),(YT-YB))
      NGS=NGK=0
      IF(DP-1.5)921,922,922
```

```
 921 NGS=NGS+1   $   GO TO 923                                              1321
 922 NGK=NGK+1                                                              1322
 923 XTA1=(XTA-XT)*(XTA-XT) $ YTA1=(YTA-YT)*(YTA-YT)                        1323
     XTB1=(XTB-XT)*(XTB-XT) $ YTB1=(YTB-YT)*(YTB-YT)                        1324
     XTA=100.*(XTA-XT)/SQRTF(XTA1+YTA1)                                     1325
     YTA=100.*(YTA-YT)/SQRTF(XTA1+YTA1)                                     1326
     YTB=100.*(YTB-YT)/SQRTF(XTB1+YTB1)                                     1327
     XTB=100.*(XTB-XT)/SQRTF(XTB1+YTB1)                                     1328
     XB=XB-XT $ YB=YB-YT                                                    1329
     AD = DISTTB $ CD=(XTA-XB)*(XTA-XB)+(YTA-YB)*(YTA-YB)                   1330
     DD=(XTB-XB)*(XTB-XB)+(YTB-YB)*(YTB-YB)                                 1331
     BD=ED=100.                                                             1332
     CD=SQRTF(CD) $ DD=SQRTF(DD)                                            1333
C    AND CENTER,ASSUME THAT MISSILE AIMED AT CENTER (XT,YT), COLUMN         1334
C    MOVING WITH 3TANKS AND 2APC AT 50 MTR INTERVAL                         1335
C                                                                           1336
     IF(TRD-AD)924,925,925                                                  1337
 924 IF(TRD-CD)1076,926,926                                                 1338
 925 IF(TRD-CD)928,1075,1075                                                1339
 928 KIND=2  $ GO TO 998                                                    1340
 926 KIND=1                                                                 1341
 998 IND=0 $  DX=DY=0   $ J=2                                               1342
 999 DX=XB  $DY=YB  $ GO TO 900                                             1343
1011 TH1=THETA                                                              1345
     IND=0 $ DX=DY=0 $ DX=XTA  $ DY=YTA  $ GO TO 900                        1346
1012 TH2=THETA $ GO TO 1016                                                 1347
1013 IND=0 $DX=DY=0 $DX=XTB  $ DY=YTB  $ J=4 $GO TO 900                     1348
1016 IF(TH1-TH2)1017,1020,1020                                              1349
1017 IF((TH2-TH1).GT.3.1416)1018,1019                                       1350
1018 CTH=6.2832-TH2+TH1          $ GO TO 1023                               1351
1019 CTH=TH2-TH1 $ GO TO 1023                                               1352
1020 IF(TH1-TH2.GT.3.1416)1021,1022                                         1353
1021 CTH=6.2832-TH1+TH2 $ GO TO 1023                                        1354
1022 CTH=TH1-TH2                                                            1356
1023 GO TO(1100,1100,1100,1100,1200),J                                      1357
1006 IF(TRD-AD)1024,1025,1025                                               1358
```

```
1024 IF(TRD-DD)1080,1026,1026
1025 IF(TRD-DD)1028,1077,1077
1028 IKIND=2 $ GO TO 1013
1026 IKIND=1 $ GO TO 1013
1100 GO TO(1101,1102),KIND
1101 ATH=ASF(AD,SINF(CTH),CD) $ C2TH=ASF(CD,SINF(ATH),TRD)
     LTH=3.1416-(ATH+C2TH) $ LA=SINF(LTH)*CD/C2TH $ K=1 $ GO TO 1060
1102 ATH=ASF(AD,SINF(CTH),TRD) $ LTH=3.1416-(ATH+CTH)
     LA=SINF(LTH)*AD/SINF(ATH) $ K=1 $ GO TO 1060
1200 GO TO(1201,1202),IKIND
1201 ATH=ASF(AD,SINF(CTH),DD) $ C2TH=ASF(DD,SINF(ATH),TRD)
     LTH=3.1416-(ATH+C2TH) $ LA=SINF(LTH)*DD/SINF(C2TH) $ K=2
     GO TO 1060
1202 ATH=ASF(AD,SINF(CTH),TRD) $ LTH=3.1416-(ATH+CTH)
     LA=SINF(LTH)*AD/SINF(ATH)    $ K=2
1060 JN=JN+1
     GO TO (1061,1066),JN
1061 IF(LA.GE.100)1062,1063
1062 TDAM=TDAM+2 $ TAPDAM=TAPDAM+1 $ L=2 $ GO TO 1070
1063 IF(LA.GE.50)1064,1065
1064 TDAM=TDAM+1 $ TAPDAM=TAPDAM+1 $ L=2 $ GO TO 1070
1065 TDAM=TDAM+1 $ L=2 $ GO TO 1070
1066 IF(LA.GE.100)1067,1068
1067 TDAM=TDAM+1  $ TAPDAM=TAPDAM+1 $ GO TO 1080
1068 IF(LA.GE.50)1069,1080
1069 TDAM=TDAM+1 $ GO TO 1080
1070 GO TO(1006,1080),K
1075 L=2 $ TDAM=TDAM+2 $ TAPDAM=TAPDAM+1 $ GO TO 1006
1076 L=1 $ GO TO 1006
1077 GO TO(1078,1079),L
1078 TDAM=TDAM+2 $ TAPDAM=TAPDAM+1 $ GO TO 1080
1079 TDAM=TDAM+1 $ TAPDAM=TAPDAM+1 $ GO TO 1080
 961 IF(PK-RANF(-1.))1080,963,963
 963 TDAM=TDAM+1
1080 XTA=XTSA $ YTA=YTSA $ XTB=XTSB $ YTB=YTSB
     RETURN
```

```
      END

      SUBROUTINE PRINT1(ATT,INUC,KSS,KGG,IA,ITD,IFC)
C
C     THIS SUBROUTINE CONTROLS PRINTED OUTPUT.
C
      TYPE INTEGER H
      COMMON/ATT1/XTI,YTI,ZTI,XAI,YAI,ZAI,XGI,YGI,ZGI,TT,TG,TTA,ISFLG,
     1          IGFLG,RNNUM,SSM,[illegible],PD,P1,[illegible],P4,P8,NNPTYP,XA,YA,
     2          ZH,XG,YG,ZGH,IMAX,[illegible],ITMAX,JTYP,TA,FT,RLT,RMAX,RMIN
     3          ,P,IKILL,ISPDY,TFT,SPT,SPA,SPB,[illegible],TF,TX,IAT,NF,NG,ZA
     4          ,ZG,HTR,TOF,P5,P6,XT,YT,ZT,INVC,INV,RC,I9
      TYPE INTEGER RNNUM
      DIMENSION XA(300),YA(300),ZA(300),ZH(300),XG(300),YG(300),ZGH(300)
     1          ,JTYP(4),TA(4),FT(4),RLT(4),RMAX(4),RMIN(4),P(100),TFT
     2          (4),SPT(25),SPA(25),SPB(25),TF(4),TX(4
     3          ),XT(300),YT(300),ZT(300),W(18,3),NX(18,4),H(5),NF(5)
     4          ,INVC(2),INV(2),ZG(300)
      DATA((W(I,1),I=1,13)=0)
      D=ATT+TTA
      IF(IFC-3)10,319,319
   10 GO TO(100,200,300),IA
  100 KSS=KSS/10 $ IF(W(1,1))101,101,102
  101 I=1 $ GO TO 103
  102 I=3
  103 W(I,1)=D$ W(I,2)=XAI$W(I+1,2)=YAI$ W(I,3)=P5 $ NX(I,1)=KSS/1000
      NX(I,2)=KSS/100-(KSS/1000)*10 $ NX(I,3)=KSS/10-(KSS/100)*10
      NX(I,4)=KSS-(KSS/10)*10 $ GO TO 317
  200 KGG=KGG/10 $ IF(W(5,1))201,201,202
  201 I=5 $ GO TO 203
  202 I=7
  203 W(I,1)=D $ W(I,2)=XGI$ W(I+1,2)=YGI $W(I,3)=P6 $ NX(I,1)=KGG/100
```

```
   10 $ NX(I,2)=KGG/100-(KGG/1000)*10 $ NX(I,3)=KGG/10-(KGG/100)*10
     NX(I,4)=KGG-(KGG/10)*10 $ GO TO 317
 300 IF(W(9,1))304,304,301
 301 IF(W(11,1))305,305,302
 302 IF(W(13,1))306,306,303
 303 IF(W(15,1))307,307,308
 304 I=9 $ GO TO 309
 305 I=11 $ GO TO 309
 306 I=13 $ GO TO 309
 307 I=15 $ GO TO 309
 308 I=17
 309 IF(KSS)314,314,312
 312 H((I+1)/2-4)=4HTANK $ NF((I+1)/2-4)=KSS
     W(I,1)=D $ W(I,2)=XTI $ W(I+1,2)=YTI $ W(I,3)=P6
     NX(I,INUC)=1 $ I=I+2
 314 IF(KGG)317,317,315
 315 H((I+1)/2-4)=4H APC $ NF((I+1)/2-4)=KGG
     W(I,1)=D $ W(I,2)=XTI $ W(I+1,2)=YTI $ W(I,3)=P6
     NX(I,INUC)=1
 317 RETURN
 319 PRINT 1000,RNNUM
     PRINT 1001
     PRINT 1002,(W(1,I),I=1,3),(NX(1,I),I=1,4),W(2,2)
     PRINT 1003,(W(3,I),I=1,3),(NX(3,I),I=1,4),W(4,2)
     PRINT 1004,(W(5,I),I=1,3),(NX(5,I),I=1,4),W(6,2)
     PRINT 1003,(W(7,I),I=1,3),(NX(7,I),I=1,4),W(8,2)
     PRINT 1006,H(1),(W(9,I),I=1,3),(NX(9,I),I=1,4),NF(1),W(10,2)
     PRINT 1007,H(2),(W(11,I),I=1,3),(NX(11,I),I=1,4),NF(2),W(12,2)
     PRINT 1007,H(3),(W(13,I),I=1,3),(NX(13,I),I=1,4),NF(3),W(14,2)
     PRINT 1007,H(4),(W(15,I),I=1,3),(NX(15,I),I=1,4),NF(4),W(16,2)
     PRINT 1007,H(5),(W(17,I),I=1,3),(NX(17,I),I=1,4),NF(5),W(18,2)
1000 FORMAT(1H1,48X,22HSUMMARY OF RUN NUMBER,I2)
1001 FORMAT(//33X,14HDAMAGE RESULTS//33X,7HELEMENT,5X,4HTIME,4X,8HPOSIT
    1ION,4X,5HRANGE,3X,9HWEAPON(S),2X,6HDAMAGE/73X,8H 1 2 3 4,3X,6HEXTE
    2NT/32X,9HSCOUT A/C)
1002 FORMAT(/42X,2H 1,F5.0,3X,4HX = ,F6.0,3X,F5.0,3X,4(I2)/52X,4HY = ,F
```

```
      16.0)
 1002 FORMAT(/42X,2H 2,F5.0,3X,4HX = ,F6.0,3X,F5.0,3X,4(I2)/52X,4HY = ,F
      16.0)
 1004 FORMAT(/32X,10HATTACK A/C/42X,2H 1,F5.0,3X,4HX = ,F6.0,3X,F5.0,3X,
      14(I2)/52X,4HY = ,F6.0)
 1006 FORMAT(/30X,13HARMOURED UNIT/40X,A4,F5.0,3X,4HX = ,F6.0,3X,F5.0,3X
      1,4(I2),7X,I2/52X,4HY = ,F6.0)
 1007 FORMAT(/40X,A4,F5.0,3X,4HX = ,F6.0,3X,F5.0,3X,4(I2),7X,I2/52X,4HY
      1= ,F6.0)
       END
          FINIS
-EXECUTE.
```

APPENDIX III

This appendix contains a complete list and card format for the input variables to the ARMREC program. Following the input description is an example and description of the program output.

It should be noted that in the input description the variable name is the name of the input variable as it is used in the program. It is not necessary to punch this name on input cards, only the values or symbols assigned to the variable names are punched on the cards.

Input data cards for program ARMREC should be prepared as indicated. The values of all input parameters are right justified in the fields assigned and the value of each variable whose name begins with I,J,K,L,M,N is a fixed point number while all others are floating point unless otherwise indicated.

Data Group	Number of cards	Variable Name	Columns of card	Description of Variable
1	1	ITL	1-2	Time length of game, less than 99.
2	1	INVC	1-10 11-20	The number of nuclear missiles carried by each helicopter in the attack section.
		INV	21-30 31-40	The number of conventional anti-armored missiles carried by each helicopter in the attack section.
3	5	SPA	1-50	The probability of sighting by the helicopter scout air-craft.
	5	SPT	1-50	The probability of sighting by the armored unit.
	5	SPG	1-50	The sighting probability of the attack helicopter air-craft.
4	1	DELV	1-10	The change in elevation of a helicopter in the pop-up mode.
		ATR	11-20	The proposed helicopter attack range.
		HTR	21-30	The helicopter acquisition time.
		TOF	31-40	The missile time of flight.

Data Group	Number of cards	Variable Name	Columns of card	Description of variable
5	1	NRUN	1-10	The number of runs to be made.
6	1	KMAX	1-10	The number of polynomials to approximate the terrain, KMAX less than or equal to 12.
		XMIN	11-20	The X coordinate of the left edge of the terrain.
		YMIN	21-30	The Y coordinate of the bottom edge of the terrain.
		XMAX	31-40	The X coordinate of the right edge of the terrain.
		YMAX	41-50	The Y coordinate of the top edge of the terrain.
		TI	51-60	The game time interval.
		TPD	61-70	The dimension of a terrain square approximated by one polynomial.
7	1	MNAX	1-10	The degree of X in the polynomial which approximates the first terrain square.
		NMAX	11-20	The degree of Y in the polynomial which approximates the terrain in the first map square.

Data Group	Number of cards	Variable Name	Columns of card	Description of variable
8	As many cards as necessary	PHI	1-80	The coefficients of the first approximating polynomial in a (4E20.10) field as obtained from the terrain simulation program.

NOTE: Data groups 7 and 8 are repeated KMAX times.

Data Group	Number of cards	Variable Name	Columns of card	Description of variable
9	1	A B C	1-20 21-40 41-60	The coefficients of the ex-potential function which estimates the overpressure. (See DASA pubs).
		W	61-80	The nuclear weapon yield in kilotons.
10	1	PK	1-10	The kill probability of a conventional anti-tank missile.
		TRD	11-20	The target radius of damage for a nuclear weapon of yield W.
11	1	NWPTYP	1-10	The number of weapon types located with armored unit.
12	1	WPNAWPNB	1-16	The name of the weapon, e.g., Red Eye.

Data Group	Number of cards	Variable Name	Columns of card	Description of variable
		JTYP	21-25	The number of weapons of this type. JTYP is proportional to the number of units in the armored unit.
		TA	26-30	An integer, the time required to acquire the target by this weapon. (Seconds).
		FT	31-35	An integer, the firing time of this weapon before reloading is required. (Seconds).
		RLT	36-40	An integer, the reloading time of this weapon. (Seconds).
		RMAX	41-50	The maximum range of this weapon in meters.
		RMIN	51-60	The minimum range in meters of this weapon.
		TFT	61-70	The total firing time available for this weapon. (Seconds).
13	3	P	1-8 9-16 17-25 etc. 72-80	The probability of a kill by this weapon for a range of 0-2500 meters in 100 meter increments.

NOTE: Data groups 12 and 13 should be repeated NWPTYP times for each of the NRUNS

Data Group	Number of cards	Variable Name	Columns of card	Description of variable
14	4	AFIELD	1-16	These four cards have the title of the run to be printed at the top of the output page.
15	1	GMAX	1-10	The maximum positive G force to be used by the aircraft.
		RC	11-20	Aircraft maximum rate of climb. (ft./min.)
		RD	21-30	Aircraft maximum rate of descent.(ft./min.)
		GS	31-40	The estimated ground speed of the scout helicopter. (knots).
		TSPD	41-50	The estimated speed of the armored unit. (mph).
		GSG	51-60	The estimated ground speed of the attack helicopter element (knots).
		TSPDM	61-70	Emergency speed of the armored unit (mph).
16	1	IMARK	1-10	This flag determines the type of flight path flown by the helicopter scout and attack elements during the reconnaissance phase.

Data Group	Number of cards	Variable Name	Columns of card	Description of variable
				NOE = nap of the earth flight
				PP = preplanned flight
				scout attack
				IMARK= 1 PP PP
				= 2 NOE NOE
				= 3 PP NOE
				= 4 NOE PP
		ISEP	11-20	The starting time separation between the scout and attack helicopter sections.
		IPOP	21-30	The minimum elapsed time between pop-ups.
		IKILL	31-40	This flag aids in determining which aircraft to fire at.
				IKILL= 1 Fire all weapons at scout aircraft if possible.
				= 2 Fire all weapons at attack aircraft if possible.
				= 3 Fire all weapons at aircraft with highest kill probability.
				= 4 Fire available weapons at both aircraft.

Data Group	Number of cards	Variable Name	Columns of card	Description of variable
17	As many cards as necessary	XA	1-20	Each of the cards of this group
			21-30	contains the X, Y and Z co-
			31-40	ordinates of the helicopter scout section in consecutive time intervals.
		ISTP	41-45	ISTP=1 on last of these data cards and is zero otherwise. It is the only variable ap- pearing on last card indicat- ing all data for scout air- craft has been read in.
18	As many cards as necessary	XG	1-20	Contains the same type of
		YG	21-30	information for the helicop-
		ZGH	31-40	ter attack section as found
		ISTP	41-45	in data group 17.
19	As many cards as necessary	XT	1-20	Contains the same type of
		YT	21-30	information for the armored
		ZT	31-40	unit as found in data group
		ISTP	41-45	17.

NOTE: Data groups 11-19 should be repeated NRUN times.

Output of Program ARMREC

Output from program ARMREC is illustrated in Figures 1, 2, 3, and 4. Figure 1 contains the land unit weapon data, the mission title and the simulation parameters for entry into ATTACK. Figure 2 shows the simulation status for each time step. Figure 3 represents the printed historical tableau containing the important interaction data. Figure 4 is a graphical presentation of the various phases of the duel simulation. Description of the variables which appear in Figures 1 and 2 is given on pages 155 and 156.

SUMMARY OF TARGET WEAPON DATA.

WEAPON TYPE	105 MM. RIFLE	.50 CAL. M.G.	37 MM. AA WPN.	INFRARED MISSLE
NUMBER OF WEAPONS	3	5	3	2
ACQUISITION TIME (SEC)	20	6	10	14
FIRING TIME (SEC)	1	10	5	1
RELOADING TIME (SEC)	5	8	2	14
MINIMUM RANGE (METERS)	450	10	100	100
MAXIMUM RANGE (METERS)	2500	1500	2000	2500

```
                  SUMMARY OF
                  THIS IS A DUMMY RUN
MISSION -.ATTACK IS PARTIALY COMPLETE
                  ALL OTHER SUBPROGRAMS ARE
                  COMPLETED


XGI  =   72100.376      YGI  =   74908.913     ZGI  =   261.171

 TG  =     119.816     GSGM  =      38.583

 P4  =           1       P3  =          10
XAI  =   73042.858      YAI  =   75543.087     ZAI  =   310.603

TTA  =     125.000      GSM  =      56.589

XTI  =   72651.157      YTI  =   75688.427     ZTI  =   267.273

 TT  =     125.000     TSPP  =       8.941

 P2  =           2       P1  =           5

         ISFLG  =             3  IGFLG  =            2
```

FIGURE 1

Land unit weapon data, mission title, and simulation parameters for entry into ATTACK

ATT = 33.000000

XAI =	72955.929	YAI =	75454.399	ZAI =	379.235
XGI =	71661.661	YGI =	76336.125	ZGI =	345.946
XTI =	73075.801	YTI =	76700.000	ZTI =	297.862

DIST. TO TGT FROM SCOUT = 1251.3555
A-G = 1460.2043

LOS, TGT TO SCOUT = 2
A-G = 1

PROB OF SCOUT SURVIVAL = 1.00000

PROB OF A-G SURVIVAL = 1.00000

ISF = 1 IGF = 1

KSS =	0	0	0	0
KGG =	0	0	0	0
TF =	0	0	0	0
IT =	0	0	0	0

ISFLG = 3 IGFLG = 2

TOK = 0 0

INVC = 4 4 INV = 2 2

ITMV = 1 ISD = 0 IGD = 0

JTYP = 3 3 2 2

FIGURE 2

Time step simulation status

SUMMARY OF RUN NUMBER, 1

DAMAGE RESULTS

ELEMENT		TIME	POSITION		RANGE	WEAPON(S) 1 2 3 4	DAMAGE EXTENT
SCOUT A/C							
	1	87	X =	73206	1218	0 0 1 0	
			Y =	75489			
	2	90	X =	73047	1194	0 0 1 0	
			Y =	75506			
ATTACK A/C							
	1	0	X =	0	0	0 0 0 0	
			Y =	0			
	2	0	X =	0	0	0 0 0 0	
			Y =	0			
ARMOURED UNIT							
	TANK	91	X =	73076	1151	1 0 0 0	1
			Y =	76700			
	TANK	92	X =	73076	1182	1 0 0 0	2
			Y =	76700			
	APC	92	X =	73076	1182	1 0 0 0	2
			Y =	76700			
		0	X =	0	0	0 0 0 0	0
			Y =	0			
		0	X =	0	0	0 0 0 0	0
			Y =	0			

FIGURE 3

Historical tableau

NOTE: Should be read as follows: One tank damaged at coordinates (X,Y) during the 91st second of game time. Weapon 1 indicates damage to tank by a nuclear missile, Weapon 2 indicates damage resulted from a conventional weapon. Aircraft damage can be inflicted by Weapons 1 through 4 which appear in the summary of target weapon data, Figure 1.

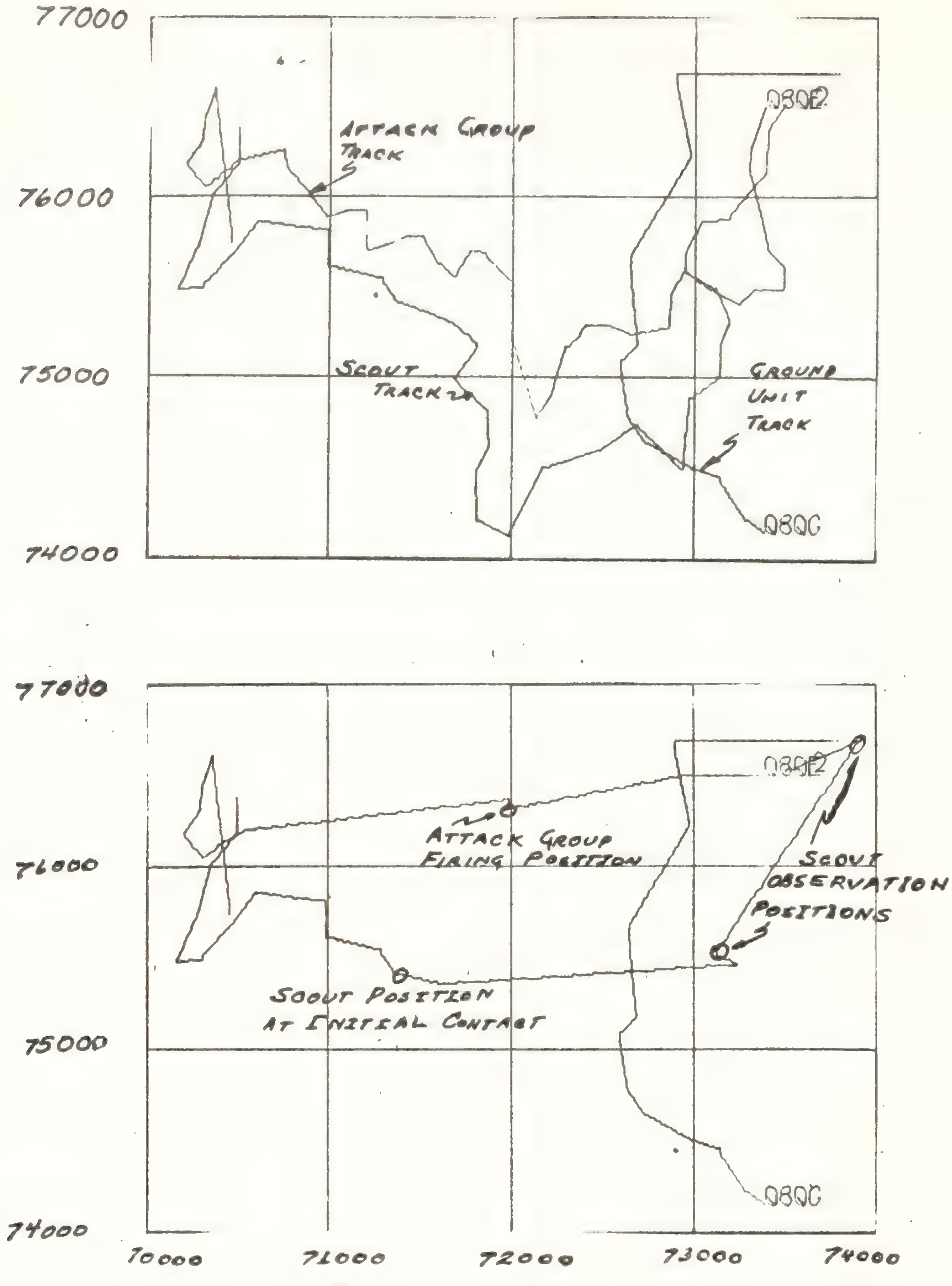


FIGURE 4

MOVEMENT OF HELICOPTER A/C AND TANKS (JOHNSON, VAN LEEUWEN)

Definitions for the variables appearing in FIGURES 1 and 2 are given below:

Variable	Definition
XAI,YAI,ZAI XGI,YGI,ZGI XTI,YTI,ZTI	Present coordinate positions of the scout helicopter section, attack helicopter section and ground unit respectively.
ISF,IGF	Results of ground unit fire on respectively the scout and attack helicopter sections. = 1 no fire = 2 fired and missed = 3 fired and hit
KSS,KGG	Depicts the ability of the land units weapons 1 through 4 to fire respectively at the scout and attack sections. = 1 can fire = 0 cannot fire
TF	The length of firing time since reloading for the land units weapons 1 through 4.
IT	The number of times that the ground units weapons have been reloaded.
ISFLG, IGFLG	Defined with comments in program ARMREC.
TOK	Estimated impact time of the attack helicopter section's missiles.

Variable	Definition
INC,INV	The number of nuclear and conventional missiles expended by attack helicopters one and two.
ITMOV	= 1 tank stopped , = 0 tank moving
ISD,IGD	The number of scout and attack helicopters damaged.
JTYP	Number of effective ground unit weapons.
P2,P4	Indicate the existence of an LOS from the scout to the ground unit and the attack section to the ground unit. = 1 no LOS = 2 LOS exists
P1,P3	The ranges from the scout section and the attack section to the ground unit, rounded up and in terms of 100 meters.

APPENDIX IV

Comments on Program TERRAIN

In order to adequately simulate the effects of terrain on the movement of the engaging units in ARMREC, the program TERRAIN, as presented by Capt. J. L. Harrison, USMC, (2) with a few minor modifications was used. With this program the elevation of any point in the area of movement of the combatants can be computed.

Terrain elevation for this model is found by using an approximating polynomial function, Z = F(X,Y), in which the coefficients of the polynomial are computed by the program TERRAIN. Use of the approximating polynomial appears to be the most accurate way of simulating terrain and eliminates the need for the large computer storage necessary when average elevation squares are used to represent terrain surfaces.

With a set of input points (Z,X,Y), the program, using the least squares method, fits a polynomial of the form Z=F(X,Y) to approximate the three dimensional surface from which the input points were taken. In this thesis, input points were obtained from terrain squares of 2100x2100 meters in order to adequately fit the 2000x2000 meter square centered inside. Eight of these squares were used to provide the maneuvering area for the scenario in ARMREC.

The surface from which the input points were taken was a mockup contour map closely resembling the Stoney Valley area of the Hunter Liggett Military Reservation. Elevations varied from 1100 to 1900 feet within this area. Polynomials of the form, $Z=\sum\sum a_{mn}X^mY^n$ which were fitted to each terrain square in the mockup area contained as many as 400 terms with the power of X and Y reaching a maximum of 20.

In order to provide the storage area required by the input data, program TERRAIN was written in FORTRAN 63 as a main program with three overlays. The main program's function is to call the separate overlays into operation. Each overlay provides the extra storage necessary by using a magnetic tape as auxiliary storage.

A facsimile of the terrain program output data which includes a printed list of the coefficients A_{nm} and a plotted 2000x2000 meter mockup terrain square appear in Figures 5 and 6 in this Appendix. The coefficients A_{nm} also appear on punched cards for input to ARMREC.

Potential users of Program TERRAIN should be aware of the changes necessary in the spacing data and parameters throughout the program when map squares larger than 1000x1000 meters are used. This is necessary so that the number of equally spaced grid lines, used in the program overlays for finding terrain elevations at equally spaced points, do not exceed 45.

When the CDC-1604 at the USNPGS computer facility is used, it is necessary that the terrain program and input data cards be read onto tape using the CDC-160. This tape is then used as the input tape on the CDC-1604. Failure to follow this procedure will yield FORTRAN 63 computer diagnostics peculiar to the system which will delete program execution.

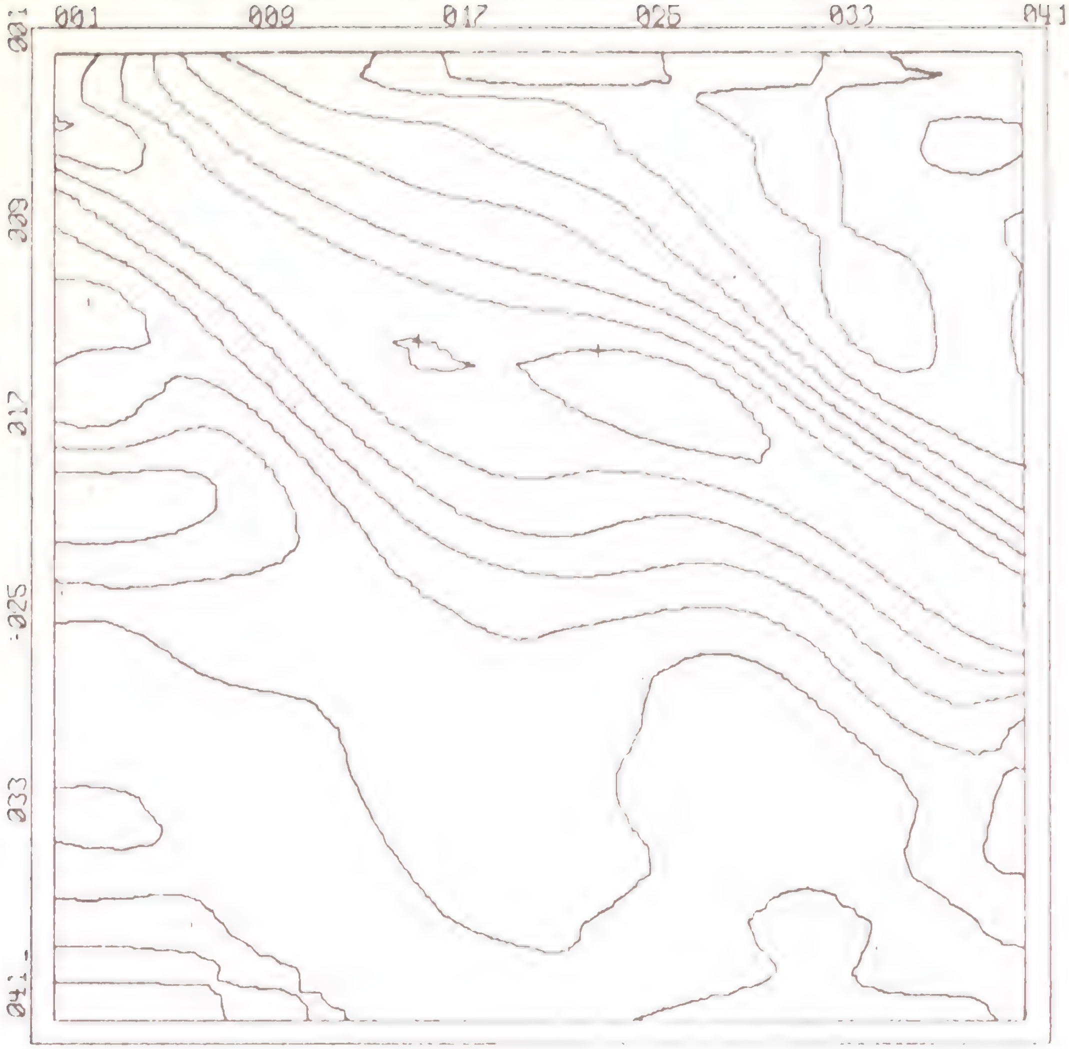


FIGURE 5

VAN LEEUWEN, JOHNSON
HLMR MOCKUP TERRAIN GRID SQUARE 5281

CCEFFICIENTS OF APPROXIMATING FCLYNOMIAL

CEGREE CF X	DEGREE CF Y 0 3 6 9 12 15 18	 1 4 7 10 13 16 19	 2 5 8 11 14 17
0	1.3235205812E 03	6.1153463628E-02	-7.2903077964E-04
	-9.0886676284E-06	1.6539701427E-08	1.5602717169E-10
	-1.6213201266E-13	-1.1350730476E-15	7.6244207965E-19
	4.4225802512E-21	-1.9981628979E-24	-1.0114873960E-26
	3.0889917657E-30	1.3986681291E-32	-2.7949186522E-36
	-1.1487293172E-38	1.3651765055E-42	5.1510330184E-45
	-2.7727314690E-49	-9.7007473202E-52	
1	-2.7191879898E-01	1.4485669925E-03	-6.8769539057E-06
	-3.8819551543E-08	1.3116543871E-10	4.8977762080E-13
	-9.5289580221E-16	-3.3786960197E-18	3.6870090048E-21
	1.3478993178E-23	-8.3779206092E-27	-3.2189376366E-29
	1.1511775414E-32	4.6527106563E-35	-9.3948519852E-39
	-3.9771493299E-41	4.1864722497E-45	1.8461950799E-47
	-7.8370136028E-52	-3.5810691393E-54	
2	3.5880351570E-04	-3.4529136088E-06	1.0115470708E-09
	1.1994126548E-10	-9.6800867394E-14	-1.6863009345E-15
	1.1093652864E-18	1.0927755537E-20	-5.3231709518E-24
	-3.9524054082E-26	1.3467812133E-29	8.6359138184E-32
	-1.9511742975E-35	-1.1636337065E-37	1.6312499313E-41
	9.4366923945E-44	-7.3329013141E-48	-4.2144166869E-50
	1.3736510764E-54	7.9477747940E-57	

FIGURE 6

INITIAL PRINT OUT OF THE COEFFICIENTS OF THE POLYNOMIAL WHICH APPROXIMATES GRID SQUARE 5281

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	20
2.	Library U. S. Naval Postgraduate School, Monterey, California	2
3.	Commandant of the Marine Corps (Code A03C) Headquarters, U. S. Marine Corps Washington, D. C. 22214	1
4.	Prof. Alvin F. Andrus Department of Operations Analysis U. S. Naval Postgraduate School, Monterey, California	4
5.	Team IV CDCEC Fort Ord, California	1
6.	CAPT Richard R. Johnson, USMC 4517 North Monticello, Chicago 25, Illinois	1
7.	CAPT Neil R. Van Leeuwen, USMC 3747 South 645 East, Salt Lake City, Utah	1

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)

U. S. Naval Postgraduate School
Monterey, California

2a. REPORT SECURITY CLASSIFICATION

Unclassified

2b. GROUP

3. REPORT TITLE

A SIMULATION OF HELICOPTER AIRCRAFT IN AN ARMED RECONNAISSANCE MODE, FOR THE CDC 1604 DIGITAL COMPUTER

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Thesis

5. AUTHOR(S) (Last name, first name, initial)

Johnson, Richard R., CAPT, USMC
VanLeeuwen, Neil R., CAPT, USMC

6. REPORT DATE

May 1966

7a. TOTAL NO. OF PAGES

161

7b. NO. OF REFS

8a. CONTRACT OR GRANT NO.

b. PROJECT NO.

c.

d.

9a. ORIGINATOR'S REPORT NUMBER(S)

9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

10. AVAILABILITY/LIMITATION NOTICES

[redacted]

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY

[redacted]

13. ABSTRACT

A model is presented which is a computer simulation of a duel involving two helicopter sections, a scout and an attack section, and an armored mobile land target. Terrain features are considered in the model by using a "least square" polynomial to represent the terrain environment. The model was constructed in an attempt to include the possible effects of terrain on tactics used by the combatants in the duel. The computer program, logic and model results are included.

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Computer Helicopter Simulation Armed Reconnaissance						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through ________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through ________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through ________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS), (S), (C), or (U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.